Oracle® Financial Data Manager Reporting

Administration Guide

Release 4.5

March 2002

Part No. A80993-02

ORACLE®

Contents

Part I   Oracle Financial Data Manager/Discoverer Integrator

1   Overview of FDM Reporting

2   Overview of FDM/Discoverer Integrator

3   Using FDM/Discoverer Integrator

4　Implementing Advanced FDM Reporting Features

5　FDM/Discoverer Integrator Error Messages

Part II　Standard Reports for Oracle Discoverer

6　Overview of the Standard Reports for Oracle Discoverer

7   Oracle Budgeting & Planning Reports

8   Oracle Customer Householding Reports

9   Oracle FDM Administration Reports

10   Oracle FDM Common Reports

11   Oracle FDM System Reports

12   Performance Analyzer Reports for the Banking and Lending Industry

13   Performance Analyzer Reports for the Insurance Industry

14   Oracle FDM Rate Manager Reports

15   Using the Oracle Risk Manager Reports

16 Oracle Transfer Pricing Reports

Part III Standard Reports for Oracle Reports

17 Overview of the Standard Reports for Oracle Reports

18 Predefined Templates

19 Oracle FDM Common Reports

20 Oracle Performance Analyzer Reports

21   Oracle FDM Rate Manager Reports

22   Oracle Risk Manager Reports

23   Oracle Transfer Pricing Reports

Index

Send Us Your Comments

Oracle Financial Data Manager Reporting Administration Guide, Release 4.5

Part No. A80993-02

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: fsdocmail@oracle.com
- FAX: (650) 506-7200   Attention: Oracle Financials Documentation Manager
- Postal service:
  Oracle Corporation
  Attention: Oracle Financials Documentation Manager
  500 Oracle Parkway
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This administration guide describes the reporting solution for the Oracle Financial Data Manager (FDM) and Oracle Financial Services (OFS) applications. The preface discusses the following:

- Intended Audience
- Organization
- Related Documents
- Conventions
- Customer Support

Intended Audience

This guide is intended for database administrators (DBAs) and others who are responsible for creating a Discoverer End User Layer (EUL) or customizing the Standard Reports. It assumes familiarity with reporting concepts.

The book parts covering the Oracle FDM/Discoverer Integrator and the Discoverer Standard Reports require knowledge of the Oracle Discoverer Administration Edition. The Oracle Reports Standard Reports book part requires knowledge of the Oracle Reports application and SQL programming.

This guide assumes that the reports and the applicable software for each solution have been installed on your client-server system.

Organization

This administration guide contains 22 chapters divided into three parts. There is also an index. The following is a brief description of the contents:

Part I, Oracle Financial Data Manager/Discoverer Integrator

Chapter 1, "Overview of FDM Reporting"

Chaper 1 discusses FDM reporting concepts.

Chapter 2, "Overview of FDM/Discoverer Integrator"

Chapter 2 discusses the purpose of the FDM/Discoverer Integrator application and how it relates to Discoverer. This chapter also provides instructions for logging into the application and describes its features.

Chapter 3, "Using FDM/Discoverer Integrator"

Chapter 3 provides instructions for using the application such as how to create Business Areas and folders. This chapter also provides information on the concepts on which FDM/Discoverer Integrator creates folders.

Chapter 4, "Implementing Advanced FDM Reporting Features"

This chapter explains the advanced features of Discoverer that enable you to manipulate FDM-related data, such as creating Annualized Rate Factors.

Chapter 5, "FDM/Discoverer Integrator Error Messages"

Chapter 5 provides a complete list of possible error messages within the FDM/Discoverer Integrator application.

Part II, Standard Reports for Oracle Discoverer

Chapter 6, "Overview of the Standard Reports for Oracle Discoverer"

Chapter 6 describes the Standard Reports and the Discoverer environment. It also provides a complete list of reports.

Chapter 7, "Oracle Budgeting & Planning Reports"

Chapter 7 explains the reports that relate to Oracle Budgeting & Planning data.

Chapter 8, "Oracle Customer Householding Reports"

This chapter provides detailed information on reports relating to Customer Householding data.

Chapter 9, "Oracle FDM Administration Reports"

Chapter 9 discusses the reports that relate to the FDM Administration application.

Chapter 10, "Oracle FDM Common Reports"

Chapter 10 explains the hierarchy report.

Chapter 11, "Oracle FDM System Reports"

This chapter provides detailed information on reports relating to system-wide activities, such as reconciling the LEDGER_STAT table.

Chapter 12, "Performance Analyzer Reports for the Banking and Lending Industry"

Chapter 12 discusses Performance Analyzer reports as they relate to the banking and lending industry.

Chapter 13, "Performance Analyzer Reports for the Insurance Industry"

Chapter 13 describes the Performance Analyzer reports available for the insurance industry.

Chapter 14, "Oracle FDM Rate Manager Reports"

This chapter provides detailed information on the multicurrency matrix.

Chapter 15, "Using the Oracle Risk Manager Reports"

Chapter 15 covers the reports that relate to Risk Manager data.

Chapter 16, "Oracle Transfer Pricing Reports"

Chapter 16 discusses the reports that relate to Transfer Pricing data.

Part III, Standard Reports for Oracle Reports

Chapter 17, "Overview of the Standard Reports for Oracle Reports"

Chapter 17 provides information on the Standard Reports within the Oracle Reports environment.

Chapter 18, "Predefined Templates"

This chapter illustrates the templates for Oracle Reports and explains how to modifify them.

Chapter 19, "Oracle FDM Common Reports"

Chapter 19 discusses the complete SQL code for the FDM Common reports.

Chapter 20, "Oracle Performance Analyzer Reports"

This chapter provides a brief description and the complete SQL code for the reports relating to the Performance Analyzer data.

Chapter 21, "Oracle FDM Rate Manager Reports"

Chapter 21 provides a brief description and the complete SQL code for the reports relating to the Rate Manager data.

Chapter 22, "Oracle Risk Manager Reports"

Chapter 22 provides a brief description and the complete SQL code for the reports relating to the Risk Manager data.

Chapter 23, "Oracle Transfer Pricing Reports"

This chapter provides a brief description and the complete SQL code for the reports relating to the Transfer Pricing data.

Related Documents

For more information about preparing for FDM reporting, using Oracle Discoverer, or using Oracle Reports, see the following documents:

- *Oracle Financial Services Installation and Configuration Guide*
- *Oracle Financial Data Manager Administration Guide*
- *Oracle Financial Data Manager Balance & Control Reference Guide*, "Transformation ID" chapter
- *Oracle Discoverer 3.1 Administration Guide*
- *Oracle Discoverer 3.1 User Guide*
- *Oracle Reports Building Reports Manual*
- *Oracle Reports Reference Manual*

Conventions

The following conventions are used in this guide:

Convention	Meaning
boldface text	Boldface type in text indicates a term defined in the text, the glossary, or in both locations.
bold monospace	Bold monospace type in text indicates information that you type in.
Italics	Italics emphasize a word or phrase.
< >	Angle brackets enclose user-supplied names (for example, <Branch Name>).
[ ]	Brackets enclose function and terminal keys. In common syntax, brackets denote one or more optional items.
{ }	Braces are used to denote variables, and in command syntax, a choice within a mandatory item.
	Example of command syntax:
	Warning: INIT file {filename} already exists.
	Example of choices:
	{EXIT|QUIT}
>	Arrowheads indicate menu paths.
■	Bullets indicate a list of items or topics.
1.	Numbered lists are used for sequential steps in completing a procedure.

Notes and Cautions

Certain information may be set off in boxes for purposes of emphasis:

- *Note* refers to interesting but incidental information about the product, or information that may be important but to a lesser degree than a *Caution* message.

- *Caution* indicates the possibility of damage to a product, system, or data.

Customer Support

Contact your project manager for the Oracle Support Services options available in your area. Besides telephone support, options may include MetaLink, which provides online access to technical libraries, patches, Technical Assistance Requests (TARs), and bugs. MetaLink is availabe at the following URL:

```
http://metalink.oracle.com.
```

Part I

Oracle Financial Data Manager/ Discoverer Integrator

The Oracle Financial Data Manager/Discoverer Integrator automates the laborious task of populating a Discoverer End User Layer (EUL). This automation reduces your time investment, and enables you to provide ready-to-use Business Areas for your users quickly and efficiently.

The material contained in this book part makes the following assumptions:

- Oracle Discoverer is installed

- Oracle FDM/Discoverer Integrator is installed

- The OFSA_EULOWNER EUL and Business Areas have been installed

- You have transformed your data using the Transformation ID

For information on the installation of these items, refer to either the *Oracle Discoverer Release 3.1 Installation Guide*, or to Chapter 13, "Installing and Configuring Discoverer," in the *Oracle Financial Services Installation and Configuration Guide*.

For information on using the Transformation ID, see the "Transformation ID" chapter in the *Oracle Financial Data Manager Balance & Control Reference Guide*.

This part contains the following chapters:

- Chapter 1, "Overview of FDM Reporting"

- Chapter 2, "Overview of FDM/Discoverer Integrator"

- Chapter 3, "Using FDM/Discoverer Integrator"

- Chapter 4, "Implementing Advanced FDM Reporting Features"

- Chapter 5, "FDM/Discoverer Integrator Error Messages"

1

Overview of FDM Reporting

Reports are necessary to analyze business profitability management. Whether you want to perform analysis on past activity or forecast future activity, you need efficient and logical access to your data. In addition, you may need to support broad distribution, remote users, or very complex reports. The FDM reporting solution provides you the tools to meet these needs.

The reporting solution involves accessing the processing data and converting it into reporting data. This chapter covers this topic and others through:

- Overview of the Oracle Financial Services Solution
- Reporting Concepts of OFS Applications
- FDM Standard Reports

1.1 Overview of the Oracle Financial Services Solution

The Oracle Financial Services solution includes Financial Data Manager (FDM) and one or more of the Oracle Financial Services (OFS) applications. The following diagram illustrates the interaction between the applications and FDM.
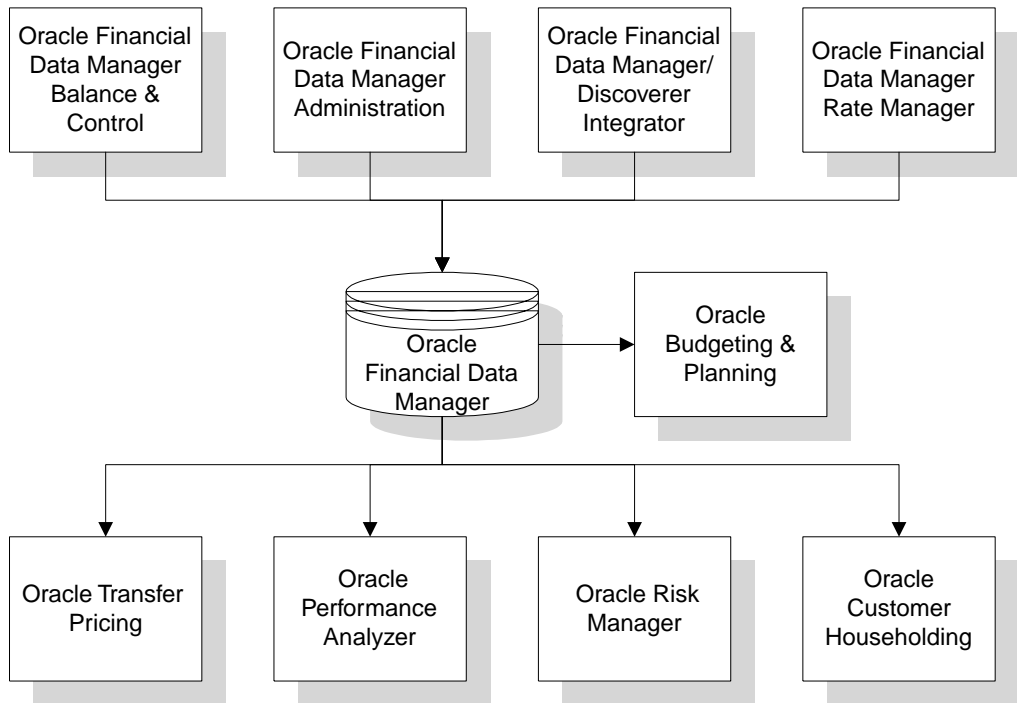
Oracle Financial Data Manager (FDM) is a standalone data warehouse with prepackaged data elements for the financial services industry. FDM is also the foundation of the OFS applications. It provides the database structures necessary to support the individual business applications.

FDM includes Oracle Financial Data Manager Balance & Control, Oracle Financial Data Manager Administration, Oracle Financial Data Manager/Discoverer Integrator, and Oracle Financial Data Manager Rate Manager.

The OFS applications form a comprehensive decision support solution that significantly enhances transfer pricing, budgeting and planning, risk management, and performance measurement functions across a financial organization.

1.2 Reporting Concepts of OFS Applications

Most of the OFS applications rely on transformed data for reporting activities. Integrations exist between FDM and the reporting products, Oracle Discoverer and Oracle Reports. This integration enables you to use either of these tools with minimal effort.

The reporting solution for FDM and the OFS applications includes a Reporting Data Mart with transformed data and Standard Reports that run against this mart. This section discusses the FDM data marts and the data transformation process. The Standard Reports are discussed in the following section.

> **Note:**   Oracle Budgeting & Planning application deviates from these reporting solutions as it has its own seeded reports accessible within the application. For more information, refer to the chapter, "Generating Reports," in the *Oracle Budgeting & Planning Reference Guide.*

1.2.1 FDM Data Marts

Data marts provide views of the integrated business intelligence provided by a larger data warehouse. The FDM data model consists of two data marts:

- Processing Data Mart
- Reporting Data Mart

The Processing Data Mart provides the required data structures for processing by the financial engines of the OFS applications.

The Reporting Data Mart provides the data structures necessary to perform reporting and analysis activities.

1.2.2 Transforming the Data

The data stored in the Processing Data Mart tables must undergo a transformation in order for you to perform reporting and analysis activities. This transformation process restructures the data and places it within Reporting Data Mart tables.

The following diagram illustrates this process:
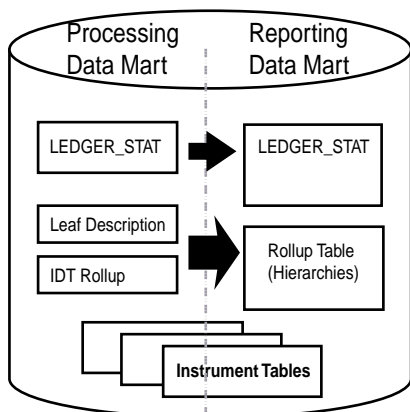
The transformation process restructures the Ledger Stat data in the Processing Data Mart into Ledger Stat tables within the Reporting Data Mart. The Leaf Description and IDT Rollup tables are transformed into Rollup tables (hierarchies). As the Instrument tables do not require transformation, they are shared between the Processing Data Mart and the Reporting Data Mart.

For more information, see the "Transformation ID" chapter in *Oracle Financial Data Manager Balance & Control Reference Guide.*

1.3 FDM Standard Reports

Standard Reports for the OFS applications are available for both Oracle Discoverer and Oracle Reports. These reports represent a wide range of capabilities of either product, and provide typical reporting needs of financial organizations. You can customize these reports or use them as starting points for your reporting needs.

Multicurrency in Standard Reports

The Standard Reports support multicurrency reporting, including the current European Monetary Union structure and legislation for the Euro. This capability enables multinational corporations to perform reporting tasks such as analyze financial performance in different currencies, report on exchange rate gains or losses, and manage exchange rate risk.

Many of the reports have a multicurrency equivalent. For example, the Balance Sheet report (balance.dis or balance.rdf) has a duplicate labeled Balance Sheet with Multicurrency (balance_mc.dis or balance_mc.rdf). The multicurrency version of the report includes all the necessary links and fields required to report on multiple currencies.

The documentation for each multicurrency report follows the documentation for the original report within their respective chapters.

2

Overview of FDM/Discoverer Integrator

Oracle Financial Data Manager (FDM)/Discoverer Integrator is a tool for populating FDM-based end user layers (EULs) in Oracle Discoverer. These EULs enable users to run reports using metadata from FDM-based applications, such as Oracle Risk Manager, Oracle Budgeting & Planning, and Oracle Performance Analyzer.

This chapter provides an overview of FDM/Discoverer Integrator through the following topics:

- Product Purpose and Process
- Oracle Discoverer
- The FDM/Discoverer Integrator Interface
- OFSA System Templates

2.1 Product Purpose and Process

The FDM/Discoverer Integrator tool provides a mechanism to convert metadata from FDM into Discoverer EUL tables.

The following diagram illustrates this process:
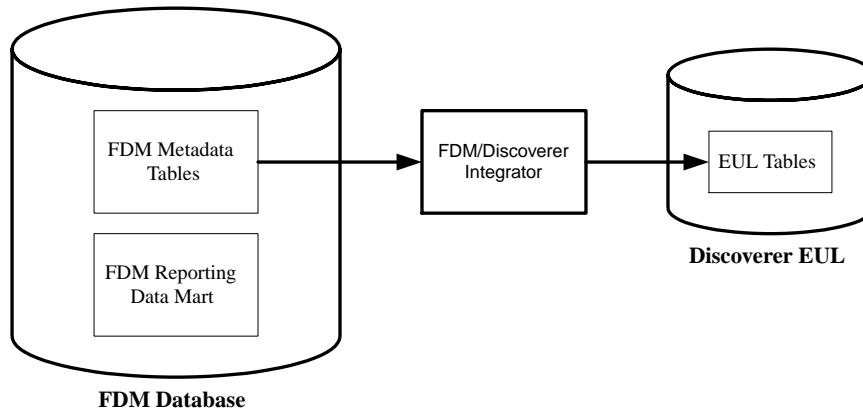
The end result is a populated EUL enabling you to perform reporting on the FDM data. Using the Discoverer EUL tables for reporting provide you with several advantages over using the FDM metadata tables directly, such as:

- The complexity of the FDM database structure is not visible.

- Column and table names appear in their long format making them more readable and identifiable.

You can use the Discoverer reporting tool to create and run reports for your organization.

A user creating a report in Discoverer uses the EUL tables to define the columns and rows of the report.
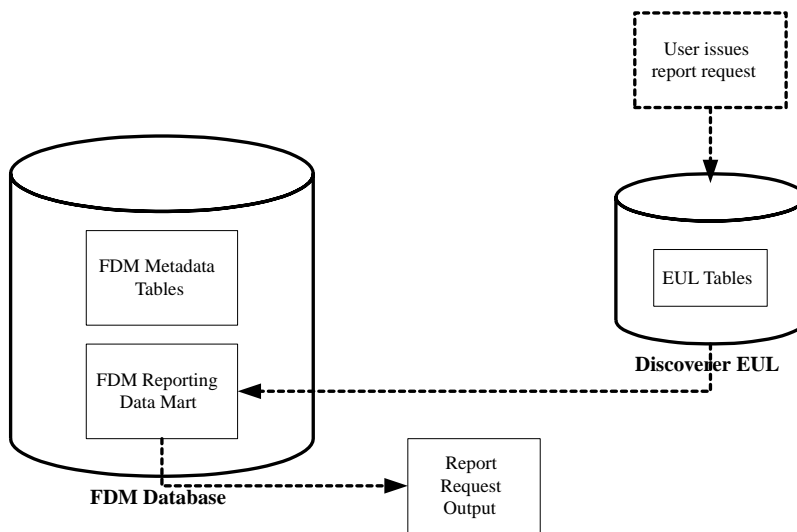
Once the user issues a report request, Discoverer retrieves and processes the data from the Reporting Data Mart. The result is a report in the specified output format.

2.2 Oracle Discoverer

Oracle Discoverer is a decision-support application that provides the ability to target and analyze specific data within your database. It uses a metadata interface to provide database access in a user-friendly manner.

2.2.1 Discoverer End User Layer (EUL)

The Discoverer End User Layer (EUL) is the metadata interface that insulates end users from database complexity. It provides an intuitive, business-focused view of the database that can be tailored to suit each user or user group.

The metadata structure of the End User Layer preserves the data integrity of the database. You cannot add, change, or delete data within the database using Discoverer. The EUL uses SQL statements to retrieve data for reporting purposes only.

You can modify the predefined EUL called OFSA_EULOWNER as the needs of your users warrant.

However, this does not restrict you from creating additional EULs through Discoverer Administration and populating them through FDM/Discoverer Integrator.

> **Note:** A schema can have only one associated EUL. Therefore, if you have multiple EULs belonging to the same database, you must have multiple schemas as well.

2.2.2 Editions of Oracle Discoverer

Oracle Discoverer 3.1 has two editions, one for end users and one for administrators.

Discoverer End User Edition

The Discoverer End User Edition is a business tool that enables users to find the data they need. Users can analyze the retrieved data, and generate answers to support important business decisions.

Discoverer Administration Edition

The Discoverer Administration Edition enables you, as the Business Systems Administrator or DBA, to create views of data (Business Areas), and maintain the EUL. While you populate the FDM-based EUL in FDM/Discoverer Integrator, you must use the Discoverer Administration Edition for advanced maintenance of the EUL.

> **Note:** You cannot use the FDM/Discoverer Integrator and the Discoverer Administration Edition on the same EUL concurrently.

For more detailed information about Discoverer, see the *Oracle Discoverer 3.1 Administration Guide.*

2.2.3 Tools Set Reference

In the set up and usage of Discoverer for FDM reporting, you must perform various tasks requiring different tools. The following table provides a list of tasks and identifies the appropriate tool to use for each task. In some cases, you can choose your preferred tool to perform a task.

Table 2–1   Tools Set Reference

Task	SQL* Plus	FDM Administration	FDM/Discoverer Integrator	Discoverer Administration	Discoverer End User
Create Reporting Data Mart Users	X	—	—	—	—
Register Reporting Data Mart Users	—	X	—	—	—
Assign/Revoke Table and Role Security Security	—	X	—	—	—
Create End User Layer	—	—	—	X	—
Assign Discoverer Permissions	—	—	—	X	—
Assign/Revoke Business Area Security	—	—	—	X	—
Create/Delete Business Areas	—	—	X	X	—
Create Folders	—	—	Reporting Data Mart Tables only	X	—
Delete Folders	—	—	X	X	—
Modify Folder/Item Properties	—	—	—	X	—
Add Items	—	—	X	X	—
Delete Items	—	—	X	X	—
Create/Delete Joins	—	—	Hierarchy only	X	—
Create Conditions	—	—	—	X	X
Create Calculations	—	—	—	X	X
Create List of Values	—	—	—	X	—
Create Alternate Sort Orders	—	—	—	X	—

Table 2–1   (Cont.) Tools Set Reference

Task	SQL* Plus	FDM Administration	FDM/Discoverer Integrator	Discoverer Administration	Discoverer End User
Define Alternate Fiscal Years	—	—	—	X	—
Activate/Deactivate Outer Join Option	—	—	X	—	—
Create/Delete/Modify Workbooks	—	—	—	—	X

2.3 The FDM/Discoverer Integrator Interface

Through the FDM/Discoverer Integrator interface, you can create, view, or delete Business Areas. The Business Areas are expandable folders that contain items, joins, and hierarchies.

As an administrator, you can manage what information is accessible by an end user through the Discoverer Administration Edition.

2.3.1 Logging in to FDM/Discoverer Integrator

Prior to logging in to the FDM/Discoverer Integrator application, you must first register your users through the FDM Administration application. For instructions on how to register users through FDM Administration, refer to the *Oracle Financial Data Manager Administration Guide.*

> **Note:**   Users created with the OFSA System Administrator 4.0 application or earlier versions, are unable to log in to FDM/ Discoverer Integrator.

To log in to the FDM/Discoverer Integrator, follow these steps:

1. Launch the FDM/Discoverer Integrator application from the Start menu or from one of the Oracle Financial Services applications and the login screen appears.

2. Select the appropriate database from the Data Source list first.

    Your username and password appear in the other prompts if they are predefined in the ofs.ini file, at which point you can click OK. Otherwise, you must complete the remaining steps.

3. Type your user name in the Username field.

4. Type your password in the Password field.

5. Click OK.

The main window of the FDM/Discoverer Integrator appears displaying any existing business areas under the EUL Root folder in a navigation tree.

If this is the first login or no Business Areas have been previously defined, the EUL Root folder is empty.

2.3.2 Menu and Toolbar

This section describes the options available in the menus and on the toolbar.

Options are enabled as appropriate. For example, the paste options are only made available after you copy a folder.

2.3.2.1 Menu Options

Most options under the menus follow the standard Windows features. The following options require further explanation:

Menu	Option	Description
File	Connect	Establishes a new connection to a database and closes the existing connection.
Edit	Paste Reference	Creates a link to the copied folder
Edit	Outer Joins	Makes all table joins exist as outer joins; a global setting
		The outer joins enable you to view all of the records in the Instrument Table, even if the corresponding code value descriptions do not exist in the individual code value description table. See the *Oracle Discoverer 3.1 Administration Guide* for more information.
Insert	Business Area > New	Creates a new Business Area
Insert	Business Area > New and Load Folders	Creates a new Business Area and activates the wizard for loading folders
Insert	Folder	Launches the wizard for loading folders
Insert	Item	Launches the wizard for adding items
Insert	Join	Launches the wizard for creating joins
Help	Contents	Access the table of contents of the Oracle Financial Data Manager Reporting Administration Guide
Help	Index	Accesses the index of the Oracle Financial Data Manager Reporting Administration Guide
Help	Oracle Financial Services Library	Accesses the library of reference guides for the Oracle Financial Services applications
Help	About Oracle Financial Data Manager/Discoverer Integrator	Provides information about the product such as the release number, the current user, and the current database

2.3.2.2 Toolbar Options

The FDM/Discoverer Integrator toolbar provides shortcuts to several menu options. Some of the icons are standard Windows features. However, a few icons are specific to the FDM/Discoverer Integrator application. These icons are as follows:

Icon	Description
	Connect
	Paste Reference
	New Business Area
	New Folder
	New Item
	New Join

2.3.3 Navigation Tree Objects

The the FDM/Discoverer Integrator navigation tree contains the following objects:

- Business Areas
- Folders
- Items
- Joins
- Hierarchies

Here is an example:
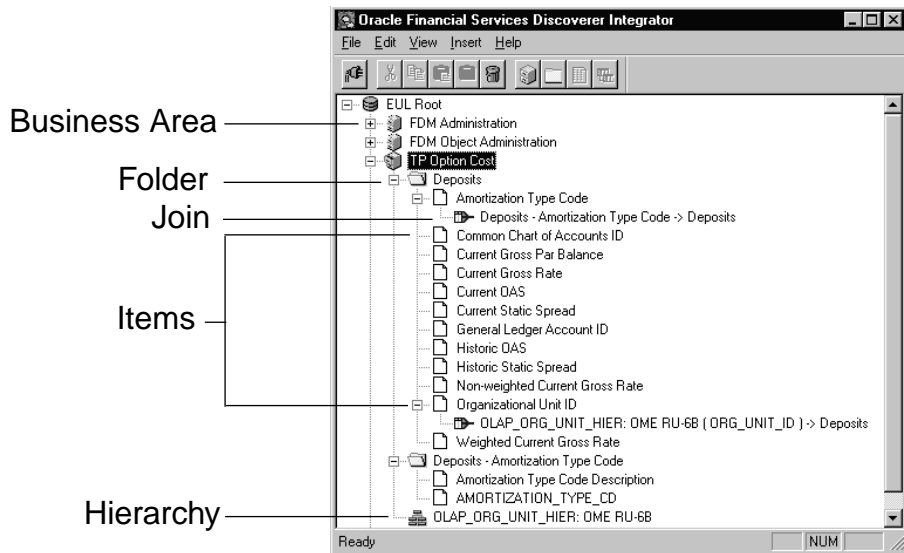
2.3.3.1 Business Areas

A Business Area is a logical grouping of tables (or views) that apply to specific reporting requirements of a user. A Business Area has one or more of the following characteristics:

- contains data from several different tables or views

- has tables or views mapped to folders

- contains many simple or complex folders

- contains folders from one or more physical databases

- is allocated to one or many database users

- enables users to access data without knowledge of the database structure

2.3.3.2 Folders

Folders store details about groups of related information. For example, one folder may contain details about the employees of a company, while another folder stores details about customers.

2.3.3.3 Joins

A join represents a relationship between tables on the basis of common values in like columns.

2.3.3.4 Items

Within folders are items that map to columns in a database table. Each item has a name, and contains specific information about the folder. For example, if a folder contains information about employees, an item might contain the name of the employee or the date of hire.

2.3.3.5 Hierarchies

Hierarchies provide the structure for the output of your data.

2.4 OFSA System Templates

The FDM/Discoverer Integrator uses two templates when processing OFSA code type columns.

- Result Scenario
- Result Type Code Description

These mandatory templates create the required code description folders within the EUL. They are visible only through the Discoverer Administration Edition within the OFSA System Templates folder.

For each template, the Visible to User setting is No. This setting hides the templates from the User Edition of Discoverer.

You should not modify or delete these templates. If you remove the template folder or the templates, errors may occur during processing. However, FDM/Discoverer Integrator will regenerate these templates upon your next login.

To improve performance, FDM/Discoverer Integrator generates an OFSA System Template for each Reporting Data Mart administrator. The template folder is named

```
OFSA System Templates - <username>
```

The individual views are:

```
Result Scenario - <username>
Result Type Code Description - <username>
```

3

Using FDM/Discoverer Integrator

The FDM/Discoverer Integrator uses wizards to generate the components of the End User Layer (EUL). This chapter covers how to build a complete and usable Business Area and use the various wizards available in the FDM/Discoverer Integrator application. The topics of this chapter include:

- Business Areas
- Managing Folders
- Managing Items
- Managing Joins

3.1 Business Areas

A Business Area is a group of folders that share a common subject, for example, Income Statements, Balance Sheets, and Market Value Analysis. Therefore, creating Business Areas requires that you first understand the reporting needs of your end users. Once you identify the requirements, you can use the New Business Area and Load Folders wizard in FDM/Discoverer Integrator to perform the creation and definition of your Business Areas. For more detailed information on Business Areas, see the *Oracle Discoverer 3.1 Administration Guide*.

3.1.1 Creating and Defining a Business Area

The creation and definition of a Business Area includes specifying all the elements you want related based on a common theme. These elements include folders, items, joins, and hierarchies.

See "Folder Concepts" later in this chapter for more information on the concepts the FDM/Discoverer Integrator application uses during folder creation.

To create and define a Business Area, perform the following steps:

1. Select Business Area > New and Load Folders from the Insert menu.

   This selection creates the Business Area and activates the Load Folder wizard.

2. Expand the System Information folder in the wizard.

3. Select the tables and columns you want in this Business Area and any desired hierarchies, and click Next.

4. Expand the Table Joins to Hierarchies folder to see all possible table joins.

5. Select the desired table joins and click Next.

6. Expand the Hierarchy Joins to Tables folder to see all possible hierarchical joins.

7. Select the desired hierarchical joins and click Next.

8. Enter a meaningful name for the Business Area and click Finish.

   You can use a maximum number of 100 characters in naming the Business Area, and you can use special characters such as $, %, &, and spaces.

   > **Note:** If you exceed the 100 character limit, the following warning message appears:
   >
   > ```
   > Internal EUL error:
   > InvalidStringLength - String length is not valid.
   > ```

The FDM/Discoverer Integrator processes the new Business Area creating folders for each table, items for each column, and all requested joins. Once the processing completes, the Business Area appears on the navigation tree. For detailed information about the selection choices in the wizard, see "Managing Folders" later in this chapter.

3.1.2 Modifying a Business Area

Since FDM/Discoverer Integrator is designed to work in conjunction with Discoverer Administration, you have limited modification abilities in FDM/Discoverer Integrator. Within a Business Area, you can add or delete objects such as folders and items. You can also modify the names of the Business Area and associated objects. For further modifications, you must use Discoverer Administration. See the *Oracle Discoverer 3.1 Administration Guide* for more information.

3.1.3 Deleting a Business Area

To delete a Business Area, complete the following steps:

1. Select the Business Area you want to delete.

2. Select Edit > Delete.

3. Click OK at the prompt to confirm the deletion.

The navigation tree appears reflecting your deletion.

3.1.4 Managing Business Area Security

Controlling access to Business Areas requires maintaining user privileges in both Discoverer Administration and Financial Data Manager (FDM) Administration. This section describes the required elements for user access and refers you to the appropriate documentation for further information and instruction.

3.1.4.1 Discoverer Administration Edition

To enable a user to access information in a Business Area, you must grant the user access through the Security option in the Tools menu in Discoverer Administration. Through this option you can:

- Grant one or more users or roles access to a specific Business Area,

- Grant access to one or more Business Areas for a specific user or role.

Additionally, you can enable a user to have Administration privileges on one or more Business Areas using the Privilege option located on the Tools menu in Discoverer Administration. Using this option, you can either:

- Grant access to one or more privileges for a specific user or role,

- Grant a specific privilege to one or more users or roles

For more information, see the *Oracle Discoverer 3.1 Administration Guide.*

3.1.4.2 Financial Data Manager (FDM) Administration

Reporting privileges encompass the ability to select data from appropriate objects within the database. You must use the FDM Administration application to maintain these privileges.

To simplify security administration, you can grant the privileges to seeded roles that provide access to all of the objects conceivably required for reporting. You can then

assign these roles to user groups or individual users as appropriate. The two roles specifically related to reporting are:

- OFDM_R_REPORT_MART

- OFDM_R_BUSINESS_PROCESS

For a complete list of seeded roles, see Chapter 11, "FDM Security," in the *Oracle Financial Services Installation and Configuration*.

If the seeded roles do not meet your security needs, you can assign privileges to individual static or dynamic objects. Static objects are permanent objects in the database, whereas dynamic objects are those tables created by Risk Manager and Transformation that output processing.

For more information on database object privileges, see Chapter 11, "FDM Security," in the *Oracle Financial Services Installation and Configuration Guide*. For more information about how to set up security within the FDM database, see Chapter 3, "Managing Security," in the *Oracle Financial Data Manager Administration Guide*.

3.2 Managing Folders

Folders map to database tables, or views of tables. Complex folders map to a combination of tables and views. Custom folders map to SQL statements.

This section discusses the concepts used in folder creation and provides instructions for adding and deleting folders.

3.2.1 Folder Concepts

This section covers various concepts applicable to the Load Folder wizard.

3.2.1.1 Table Classifications

The following table provides a list of the FDM/Discoverer Integrator table types with the associated OFSA classification and table classifications.

FDM/Discoverer Integrator Table Type	OFSA Classification	Table Classification
Hierarchy	Rollup	250
Instrument	Instrument	20
Ledger Stat Reporting	Ledger Stat	220
Reporting Rates	Reporting Rates	460
Reporting System Tables	Reporting System Tables	450
Risk Manager Result Detail	Cash Flow Result Detail	420
Risk Manager Result Detail	Gap Result Detail	420
Risk Manager Result Master	Result Master	150
Risk Manager VaR	VaR	182
Risk Manager EaR	EaR	185
User Defined	User Defined	320

For more information on the classifications of the tables, see the *Oracle Financial Data Manager Administration Guide*.

3.2.1.2 Registering the Tables

Registering the tables means creating the table instances in the appropriate EUL. The following table depicts the source of the required information:

Discoverer Attribute	Source
Name	OFSA_TABLES Display Name
Description	OFSA_TABLES Description
Visible to User	OFSA_TABLES Display Flag
Object	OFSA_TABLES Table Name

3.2.1.3 Registering the Columns

Registering the columns means creating the column instances in the appropriate EUL. The following table depicts the source of the required information:

Discoverer Attribute	Source
Name	OFSA_TAB_COLUMNS Display Name
Description	OFSA_TAB_COLUMNS Definition
Visible to User	OFSA_TAB_COLUMNS Display Flag except for rate columns, which are set to No.
Item Class	None
Date Hierarchy	See "Non-Date Hierarchies"
Default Position	For financial element columns - data point
	For date columns - top
	For Leaf ID columns - side
	For all other columns - top
Default Aggregate	Use the Discoverer defaulting mechanism, except for rate fields, which are Detail.
Sequence	Based on the sequence in the OFSA_TAB_COLUMNS table
Heading	OFSA_TAB_COLUMNS Display Name
Format Mask	Balance Fields - L999G999G999G999G999D999
	Date Fields - Discoverer Standard format
	Rate Fields - 99999D9999PR
	All Others - Default
Alignment	General
Word Wrap	No

3.2.1.4 Instrument and Ledger Stat Code Values

For each selected instrument table column, FDM/Discoverer Integrator checks to see if any of the columns are defined as CODE in the OFSA_TAB_COLUMN table where the OFSA_DATA_TYPE_CD = 3.

For any CODE columns, FDM/Discoverer Integrator does the following:

1. Creates a new folder based on the individual code value description table for that code type. The name of this new folder is `<Instrument folder name> - <item name>`.

    Example: `Mortgage Backed Securities - Amortization Type Code`

2. Creates a join between the code column of the newly created folder to the code column in the instrument table folder. The name of the join is `<parent folder name> > <child folder name>`. The parent folder is the name of the folder based on the individual code value description table. The child folder is the folder name of the instrument table.

    Example: `Deposits - Amortization Code > Deposits`

    Note: If the Outer Joins option is enabled, the Join Relationship property is changed to One to Many (1:n). This property enables you to view all Instrument table records, even if the corresponding code value descriptions do not exist in the individual code value description table.

    ---

3. Sets the Visible to User property to No for the code value column.

4. Sets the name of the Column Description item to `<item name>` Description, where the item name is the name of the code column in the instrument table.

    Example: `Amortization Type Code Description`

FDM/Discoverer Integrator also follows this process for the Consolidation Code column in Ledger_Stat.

3.2.1.4.1 Instrument Example If the selected folder is Deposits, and this folder contains a code value column called AMORTIZATION_TYPE_CD, then a new folder is created based on the individual code value description table.

FDM/Discoverer Integrator then does the following:

1. Modifies the folder so the users can see only the description

2. Creates a join between the new folder and the deposits folder, based on the code value column.

3.2.1.5 Non-Date Hierarchies

If the table is a hierarchy type table, FDM/Discoverer Integrator automatically creates a hierarchy based on the table, after the load of the table definition completes. At each hierarchy level, the value, sequence, and description columns are transferred from the original rollup table.

3.2.1.5.1 LEAF_DESCRIPTION Column The column at the lowest level of the hierarchy is the LEAF_DESCRIPTION column. The next lowest column is the LEVEL_01_DESC column. The next lowest column is the LEVEL_02_DESC column. Since the number of columns in the hierarchy table is dynamic, this level numbering continues until all columns are used.

For each non-date hierarchy join selected by the user, FDM/Discoverer Integrator creates a join between the hierarchy and the appropriate column. The name of the join is `<Parent Folder Name> > <Child Folder Name>`.

Example: `Branch Office Hier (ORG_UNIT_ID) -> Deposit`

3.2.1.5.2 Optional Filters In addition, for each selected non-date hierarchy, FDM/Discoverer Integrator creates an optional filter to eliminate all orphan nodes in the hierarchy. The Orphan Filter is based on the LEVEL_01_VALUE item in the hierarchy table.

3.2.1.6 Non-Weighted and Balance-Weighted Rate Columns

Storage of rate values occurs in one of two ways in the FDM database:

- Non-weighted Rate Columns
- Balance-weighted Rate Columns

You can load any rate columns, as shown in FDM/Discoverer Integrator. When loading a Calculated Non-Weighted or Balance-Weighted Rate Column, you must load the corresponding balance column as defined in the PROPERTY_COLUMN field of the OFSA_COLUMN_PROPERTIES where the COLUMN_PROPERTY_CD is 20. An example of such a related field is Current Gross Par Balance.

The default balance column associated with a rate column is predefined during the Database Upgrade Process. You can customize the PROPERTY_COLUMN field as needed. For more information, refer to the *Oracle Financial Services Installation and Configuration Guide.*

3.2.1.6.1 Non-Weighted Rate Columns Non-weighted rates are the true rates for a record. For example, if a loan record contains a rate value of 7.03, then the true rate is 7.03%. These types of rate fields are standard in the Instrument table, Portfolio Instrument table, and User-defined table. You can use this field in reports that look at detail records. However, most reports look at aggregated data. To aggregate rate data, it must be weighted by a balance column. For example, the Current Net Rate needs to be weighted by its related field Current Gross Par Balance.

For each rate column selected, FDM/Discoverer Integrator generates three rate items:

- **Non-weighted Rate Item:** The Non-weighted Rate item has the Visible to User property set to No. The item name is the name of the selected rate column prefixed by the text Non-weighted, for example, Non-weighted Current Net Rate. The formula is equal to the rate as stored in the OFSA instrument table. For example, the formula for Non-weighted Current Net Rate column is:

  COMMERCIAL_LOAN.CURRENT_NET_RATE

- **Weighted Rate Item:** The Weighted Rate item has the Visible to User property set to No. The item name is the name of the selected rate column prefixed by the text Weighted, for example, Weighted Current Net Rate. The formula is equal to the non-weighted rate column multiplied by the appropriate balance column. For example, the formula for Weighted Current Net Rate column is:

  Non-weighted Current Net Rate*Current Gross Par Balance

- **Calculated Rate Item:** The Calculated Rate item has the Visible to User property set to Yes. The name is the name of the selected rate column, for example, Current Net Rate. The formula is equal to the weighted rate column divided by the appropriate balance column.

  In addition, FDM/Discoverer Integrator wraps a DECODE statement around the calculation of the Calculated Rate item that outputs a zero in the case that the balance column is a zero. For example, the formula for the calculated Current Net Rate item is:

  DECODE(SUM(Current Gross Par Balance),0,0,(SUM(Weighted Current Net Rate)/ SUM(Current Gross Par Balance))

3.2.1.6.2 Balance-Weighted Rate Columns Balance-weighted rates are the true rates weighted by a related balance for a record. For example, if a loan record contains a weighted rate value of 80.4 and the associated balance for the rate field is 200, then the true rate is 4.02% (80.4/200). These types of rate fields are standard in the transformed LEDGER_STAT and Risk Manager Result Detail tables. This field can be used in reports that look at summary records. However, users may want to see the true rates in some cases. For example, the weighted WARM rate of the Risk Manager Result Detail table needs to be converted by dividing it by its related field, for example, End Balance.

For each balance-weighted rate column selected, FDM/Discoverer Integrator generates two rate items:

- **Weighted Rate Item:** The Weighted Rate item has the Visible to User property set to No. The item name is the name of the selected rate column prefixed by the text Weighted, for example, Weighted WARM. The formula is equal to the rate as stored in the transformed table. For example, the formula for the Weighted WARM column of transformed Risk Manager Result Detail table OFS_RPT_RES_DTL_10522_C is:

  OFS_RPT_RES_DTL_10522_C.WARM

- **Calculated Rate Item:** The Calculated Rate item has the Visible to User property set to Yes. The item name is the name of the selected rate column (for example, WARM). The formula is equal to the weighted rate column divided by the appropriate balance column. In addition, FDM/Discoverer Integrator wraps a DECODE statement around the calculation of the Calculated Rate item that outputs a zero in the case where the balance column is a zero. For example, the formula for the calculated WARM item is:

  DECODE(SUM(End Balance),0,0,(SUM(Weighted WARM)/SUM(End Balance))

  > **Note:** Transformed Risk Manager rates are stored as true rate percentage points, such as 7.5, while transformed Ledger Stat rates are stored as weighted rate ratios, such as 0.075. This storage method implies a different interpretation of the rate value between Risk Manager and Ledger Stat reports.
  >
  > To standardize calculation of rates, neither Risk Manager rates nor Ledger Stat rates are divided by 100 in the calculated rate formula.

3.2.1.7 Risk Manager Result Detail - Scenario Number Codes

For any SCENARIO_NUM columns loaded with Risk Manager Result Detail tables, FDM/Discoverer Integrator does the following:

1. Creates a new folder based on the OFSA_RESULT_SCENARIO table. The name of this new folder is `<Result Detail folder name> - Scenario Description`, where the Result Detail folder name is the name of the result detail folder (which can be different from the result detail table name).

   Example: `Result Detail 100457 - Scenario Description`

2. Joins the two Scenario Number items of the newly created folder and the result detail table folder. The name of the join is `<parent folder name> > <child folder name>`, where the child folder is the folder name of the result detail table and the parent folder is the name of the folder based on the OFSA_RESULT_SCENARIO table.

   Example: `Result Detail 100457 - Scenario Description > Result Detail 100457`

3. Creates a mandatory filter on the new folder. The filter name is the same as the new folder's name. The filter is as follows:

   `RESULT_SYS_ID = <Process System Id Number of the result detail table>`

4. Sets the Visible to User property to No for the following items in the new folder:

   - Scenario Num
   - Result Sys ID

5. Sets the name of the Description item to Result Scenario Description

3.2.1.8 Risk Manager Result Master - Scenario Number Codes

For any SCENARIO_NUM columns loaded with Risk Manager Result Master tables, the FDM/Discoverer Integrator does the following:

1. Creates a new folder based on the OFSA_RESULT_SCENARIO table. The name of this new folder is `<Result Master folder name> - Scenario Description` where the Result Master folder name is the name of the Result Master folder (which can be different from the Result Master table name).

   Example: `Result Master - Scenario Description`

2. Joins the two Scenario Number items from the newly created folder and the Result Master table folder. This join also joins the two Result System ID items from the two folders. The name of the join is `<parent folder name> > <child folder name>` where the child folder is the folder name of the Result Master table and the parent folder is the name of the folder based on the OFSA_RESULT_SCENARIO table.

   Example: `Result Master - Scenario Description > Result Master`

3. Sets the Visible to User property to No for the following items in the new folder:

   ■ Scenario Num

   ■ Result Sys ID

4. Sets the name of the Description item to "Result Scenario Description."

3.2.1.9 Risk Manager Result Type Codes

For any RESULT_TYPE columns loaded with Risk Manager Result Detail tables, the application:

1. Creates a folder based on the OFSA_RESULT_TYPE_DSC folder. The name of this folder is `<Result Detail Folder Name> - Result Type Description`.

2. Sets the Visible to User property to No for the Result Type Code item.

3. Joins the Result Type Code item in the Result Type Description folder to the Result Type item in the result detail folder. The name of the join is `<Parent Folder Name> > <child folder name>` where the child folder is the folder name of the result detail table, and the parent folder is the name of the folder based on the OFSA_RESULT_TYPE_DSC table.

   Example: `Result Detail 10047 - Result Type Description > Result Detail 100457`

3.2.2 Adding a Folder

To add a new folder in a Business Area, follow these steps:

1. Select the Business Area to which you want to add the folder.

2. Select Insert > Folder to activate the Load Folder wizard.

3. Expand the System Information folder in the wizard.

4. Select the tables and columns you want in this Business Area and any hierarchies and click Next.

   The wizard prompts you for the joins that are applicable to the selected tables. If your selection includes non-hierarchy tables, the next page of the wizard prompts you to select hierarchies to join to tables. Likewise, if your selection includes hierarchy tables, the third page of the wizard prompts you to select tables to join to hierarchies.

5. Expand the Table Joins to Hierarchies folder to see all possible table joins.

6. Select the desired table joins and click Next.

7. Expand the Hierarchy Joins to Tables folder to see all possible hierarchical joins.

8. Select the desired hierarchical joins and click Next.

9. Click Finish.

The folder appears in the selected Business Area.

3.2.3 Deleting a Folder

To delete a folder, complete the following steps:

1. Select the Business Area you want to delete.

2. Select Edit > Delete.

3. Click OK when prompted to confirm the deletion.

The Business Area on the navigation tree reflects the change immediately.

3.3 Managing Items

An item is a database column. This section provides instructions for adding and deleting items in FDM/Discoverer Integrator.

3.3.1 Adding Items

To add an item to a folder, complete the following steps:

1. Select the folder in which you want to add the item.

2. Select Insert > Item.

   Any table column that have not yet been loaded appear in the first page of the wizard.

3. Highlight the items you want to add.

4. Click Finish.

If any of the columns selected are leaf columns, the wizard displays all possible joins for those columns. From this list, you can select the desired joins which link the leaf columns to the appropriate hierarchy tables.

3.3.2 Deleting Items

To delete an item, complete the following steps:

1. Select the item you want to delete.

2. Select Edit > Delete.

If the selected column contains a join, a warning message appears informing you that the associated join will be deleted if the Delete action is continued.

3. Click OK to confirm the deletion.

The item no longer appears on the navigation tree.

3.4 Managing Joins

A join is a link between two folders (database tables), based on a common item (database column), or between a table and a hierarchy. This section provides instructions for adding and deleting joins.

You can create a join using either leaf columns or hierarchies by using one of the following join methods:

- **Column To Hierarchies:** The column to hierarchy method assumes you want to select a leaf column and join the leaf column to one or more hierarchies. Once you select a leaf column, you can add joins.

- **Hierarchy To Tables:** The hierarchy to table method assumes you want to select a hierarchy table and join it to one or more data tables. Once you select a hierarchy table in the main interface, you can perform any of the above functions to add joins.

Creating joins by either method produces the same result. Both join types are one to many joins where the hierarchy table is the master table.

3.4.1 Adding Joins

To add a join, complete the following steps:

1. Select the leaf column or hierarchy you want to join from.
2. Select Insert > Join.
3. Select the desired join from the list of possible joins.
4. Click OK.

The new join appears on the navigation tree.

3.4.2 Deleting Joins

To delete a join, complete the following steps:

1. Select the join you want to delete.
2. Select Edit > Delete.
3. Click OK to confirm the deletion.

The join no longer appears on the navigation tree.

4

Implementing Advanced FDM Reporting Features

You can implement features that provide more advanced reporting capabilities using both the FDM/Discoverer Integrator and the Discoverer Administration Edition applications.

This chapter provides instruction on:

- Defining an Alternate Fiscal Year
- Creating a List of Values (LOV)
- Creating an Alternate Sort
- Creating an Annualized Rate Factor
- Registering PL/SQL Functions for Multicurrency

4.1 Defining an Alternate Fiscal Year

Users may want to report their data based on an Alternate Fiscal Year, which is a fiscal year that does not start on January 1. To implement Alternate Fiscal Year, you must do the following:

- Finding the Fiscal Year Modifier
- Creating a Fiscal Year Hierarchy Template
- Finding the Date for the Fiscal Quarter and Year
- Creating the Date Items
- Merging the Two Date Hierarchies
- Cleaning Up

> **Note:** Some organizations may have fiscal periods that do not consist of 12 months. For example, it is standard for Japanese institutions to use six month fiscal years. Discoverer applications only supports fiscal years with periods equaling 12 months.

4.1.1 Finding the Fiscal Year Modifier

You need the fiscal year modifier in order to find the fiscal year and quarter. To find the fiscal year modifier, complete the following steps:

1. Open SQL*Plus or any database query tool using a user ID that has access to the FDM schema.

2. Enter the following SQL statement:

    ```
    SELECT * FROM OFSA_FISCAL_YEAR_INFO
    ```

    If the fiscal period column is 12, you can continue to the next step.

    If the fiscal period column provides a value other than 12, you either do not have an alternate fiscal year, or you have one that is not supported by Discoverer.

3. Subtract the value in the start month column from 13 to get the fiscal year modifier.

By keeping track of the resulting number, you only have to perform this step once.

4.1.2 Creating a Fiscal Year Hierarchy Template

Discoverer Administration uses fiscal year templates to create date hierarchies and the corresponding items. The template specifies the levels in the date hierarchy to be used (for example, Year, Month) and the format for each level (for example, "YYYY" or "YY").

You can customize the following steps. See the *Oracle Discoverer 3.1 Administration Guide* for more information on creating date hierarchy templates.

> **Note:** You only need to perform this step once.

To create a standard fiscal year hierarchy template, complete the following steps:

1. Select the New Hierarchy button on the toolbar.
2. Choose Date Hierarchy.
3. Click Next.
4. Select New Format.
5. Enter the following in the box labeled "New Format": `"FY" RR`
6. Click OK.
7. Expand the Year folder.
8. Double-click on the "FY" RR item.
9. Select New Format.
10. Choose Quarter from the date format group drop down list.
11. Enter the following in the New Format box: `"FQ"Q-RR`
12. Click OK.
13. Expand the Quarter folder.
14. Double-click on the "FQ"Q-RR item.
15. Expand the Month folder.
16. Double-click on any of Month items.
17. Expand the Day folder.
18. Double-click on any of the Day Items.
19. Click Next. **Do nothing on this page**.
20. Click Next. **Do not** set as the default date hierarchy.
21. Name the hierarchy template "Fiscal Date Hierarchy Template."
22. Click Finish.

4.1.3 Finding the Date for the Fiscal Quarter and Year

To create a calculated item that provides the date used to find fiscal quarter and year, complete the following steps:

1. Right-click on the folder that contains the date item.

2. Select New Item and the new item interface appears.

3. Select Functions.

4. Expand Date in the list of function types.

5. Double-click on ADD_MONTHS. This adds the function to the calculation.

6. Click Items.

7. Expand the folder that contains the date item.

8. Double-click the date item.

9. Enter a comma (,) and press the space key.

10. Type in the fiscal year modifier, located in the previous section "Find Fiscal Year Modifier."

11. Rename the calculated item to the following:

    ```
    "Fiscal " || date item name.
    ```

12. Click OK.

4.1.4 Creating the Date Items

Every Hierarchy, including date hierarchies, uses items in a folder as the basis of the rollup. For date hierarchies, these items are typically derivatives of the original date item. When the user applies a date hierarchy template to a date item, Discoverer Administrator automatically creates these derived date items.

To create these items, complete the following steps:

1. Right-click on Fiscal date item from the previous section "Find the Date for the Fiscal Quarter and Year."

2. Select Properties… from the menu.

3. Select Fiscal Year Date Hierarchy Template from the drop down list next to the Date Hierarchy property.

4. Click OK.

The application creates new date hierarchy items based on the fiscal date and then creates a hierarchy on the new items.

4.1.5 Merging the Two Date Hierarchies

At this point, you should have two date hierarchies: one based on the actual date item and one based on the fiscal date item. To merge the two date hierarchies into one date hierarchy, complete the following steps:

1. Select the hierarchies tab in the main window.

2. Expand the business area that you are working in.

3. Right-click on the hierarchy based on the original date column.

4. Select Edit Hierarchy from the menu.

5. Click the Items tab.

6. Double-click the Year and quarter items in the right pane.

7. Expand the folder that contains the fiscal date item in the left window, which contains the Hierarchy items.

8. Double-click on the fiscal year and quarter items. You created these items during the previous section.

9. Promote the quarter item to the top in the right pane.

10. Promote the year item to the top.

11. Click the Name tab.

12. Rename the hierarchy as appropriate, if desired. The recommended name is
    `<Folder Name> || <Date Item Name>`

13. Click OK.

4.1.6 Cleaning Up

The final step in working with alternate fiscal years is cleaning up the remaining hierarchy and extra data columns. To delete the unwanted fiscal year hierarchy and the extra date columns, and hide columns the user should not see, complete the following steps:

1. Click the Hierarchies tab in the main window.

2. Select the hierarchy based on the fiscal year column.

3. Press the Delete key to delete the hierarchy and the unused fiscal date items.

4. Click the Data tab in the main window.

5. Right-click on the fiscal date item.

6. Select Properties from the menu.

7. Change the Visible to User property to No.

8. Click OK.

9. Select the item named `<Date Item> || "Year"`.

10. Press the Delete key.

11. Select the item named `<Date Item> || "Quarter"`.

12. Press the Delete key.

4.2 Creating a List of Values (LOV)

A list of values (LOV) is the set of unique values of the item. The values the item class references correspond to those found in a database column. For example, if the database contained 4 occurrences of widgets, 28 occurrences of bolts, 34 occurrences of fan belts, 90 occurrences of gaskets, and 49 occurrences of brackets, the item class would produce a list of five distinct values: widgets, bolts, fan belts, gaskets, and brackets.

End users use LOV to refer to values in the database and to apply conditions and parameter values. The FDM/Discoverer Integrator does not automatically create LOV. To generate these lists, the Item Class Wizard of the Discoverer Administration Edition is used to create LOV and assign it to an item.

To create a new Item Class for LOV, complete the following steps:

1. Log into Discoverer Administration Edition.

2. Open an existing Business Area.

   The Data tab page appears.

3. Click the Item Classes tab.

4. Select Insert > Item Class to launch the Item Class Wizard.

5. Select the LOV option under Item Class Attributes.

6. Click Next.

7. Select the item that generates LOV, such as Deposits.Account Officer.

8. Click Next.

9. Select the items that use this item class, for example, RM Result Details.

10. Click Next.

11. Enter the name and description of the item class, for example, Account Officer LOV

12. Click Finish. This creates a new item class that contains a list of values and the items that use this item class.

13. Click the Data tab.

A LOV is now created for the items selected in step 7.

4.3 Creating an Alternate Sort

Alternate Sorts are used where users want one item to be sorted based on the values in a separate column. For example, the Balance Sheet statement is traditionally sorted in the order of Assets, Liabilities, and Equities. If you sort these values alphabetically, they would appear as Assets, Equities, and Liabilities.

The FDM/Discoverer Integrator provides a sequence column for Alternate Sort. Users can sort a report using the order specified in the sequence column. The latter is stored in the hierarchy table that was transformed from the original OFSA Rollup table. At each level of the hierarchy table, the sequence column is LEVEL_nn_SEQ, where nn is the level number.

An Alternate Sort only affects the sort sequence of items, and is invisible to users, except in its sorting effect. It requires a folder that contains the sequence (for example, Level 1 Sequence) and LOV items (for example, Level 1 Description). This folder is then automatically joined into the reporting query at run time.

The Item Class Wizard of the Discoverer Administration Edition is used to create an Alternate Sort. There are two methods available to create an Alternate Sort:

- Create a new Item Class and select both LOV and Alternate Sort in the first step of the wizard.

- Edit an existing Item Class and add the Alternate Sort.

To remove Alternate Sort without deleting the Item Class, click on the Alternate Sort item to deselect it.

4.4 Adding an Alternate Sort

To create a new Item Class for Alternate Sort, complete the following steps:

1. Log into Discoverer Administration Edition.

2. Open an existing Business Area.

   The Data tab page appears.

3. Click the Item Classes tab.

4. Select Insert > Item Class to launch the Item Class Wizard.

5. Select the Alternate Sort option under Item Class Attributes.

6. Click Next.

7. Select the item that generates LOV, for example, Level 1 Description.

8. Click Next.

9. Select the item that contains the Alternate Sort sequence, for example, Level 1 Sequence.

10. Click Next.

11. Select the items that use this item class, for example, Ledger Stat and Mortgages.

12. Click Next.

13. Type the name and description of the item class, for example, Financial Statement Alternate Sort.

14. Click Finish. This initiates the creation of a new item class which contains:

    ■ List of values sorted by alternate sort key

    ■ Items using this item class.

15. Click the Data tab again.

An LOV is now created for the items selected in step 7 above. For example, the LOV for Level 1 Description now contains ASSETS, LIABILITIES & EQUITY, and Orphan.

> **Note:** There is no specific relationship between the LOV and Alternate Sort features except that they are both implemented using the Item Class mechanism.

4.5 Creating an Annualized Rate Factor

The following instructions explain how to create an annualized rate factor using one of the following day-count methods:

- 30/360
- 30/365
- 30/Actual
- Actual/360
- Actual/365
- Actual/Actual

Typically, an end user utilizes the calculated annualized rate factor in the Discoverer End User Edition by dividing this factor with any given value to obtain the annualized value. Since the calculation of the annualized rate factor is based on the As of Date, users must create an "As of date" parameter within their workbook.

To create an annualized rate factor using an "As of Date" parameter in your workbook, complete the following steps:

1. Use Oracle Discoverer User Edition to open the workbook in need of annualized rate factor calculation.

2. Choose Tools | Parameters… from the main menu.

3. Select New…

4. Select <None> from the "For Item" list.

5. Name the Parameter, such as As of Date.

6. Enter a prompt, such as As of Date (dd-mon-yyyy).

7. Enter a description.

8. Uncheck the checkbox for "Let user enter multiple values".

9. Enter a default value and click OK.

10. Click OK.

11. Click the Edit Sheet icon in the toolbar.

12. Click the Calculation tab.

13. Click New in the Calculations screen.

14. Enter one of the following Rate Factor calculation formula in the Calculation box, depending on day count policy in your organization:

■ 30/360 Rate Factor Calculation

   (30/360)*100

■ 30/365 Rate Factor Calculation

   (30/365)*100

■ 30/Actual Rate Factor Calculation

   30/
   (TO_DATE('31-dec-'||TO_CHAR(TO_DATE(:As of
   Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')-TO_DATE('01-jan-'||TO_
   CHAR(TO_DATE(:As of Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')+1)*100

■ Actual/360 Rate Factor Calculation

   (TO_DATE(:As of Date,'dd-mon-yyyy')-TO_DATE('01-jan-'||TO_CHAR(TO_
   DATE(:As of Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')+1)/360)*100

■ Actual/365 Rate Factor Calculation

   (TO_DATE(:As of Date,'dd-mon-yyyy')-TO_DATE('01-jan-'||TO_CHAR(TO_
   DATE(:As of Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')+1)/365*100

■ Actual/Actual Rate Factor Calculation

   ((TO_DATE(:As of Date,'dd-mon-yyyy')-TO_DATE('01-jan-'||TO_CHAR(TO_
   DATE(:As of Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')+1)/

   (TO_DATE('31-dec-'||TO_CHAR(TO_DATE(:As of
   Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')-TO_DATE('01-jan-'||TO_
   CHAR(TO_DATE(:As of Date,'dd-mon-yyyy'),'yyyy'),'dd-mon-yyyy')+1)*100)

15. Change the name. For example, "<Rate Factor Name>".

16. Click OK.

17. Click New in the Calculations screen.

18. Enter the following formula in the Calculation box:

    ```
    <Item Name>/"<Rate Factor Name>"*100
    ```

    Example: `Deposits.Current Gross Par Balance/"Actual/Actual Rate Factor".`

19. Change the name and click OK.

20. Click OK.

4.6 Registering PL/SQL Functions for Multicurrency

In addition to the standard PL/SQL functions provided by Oracle, you can create custom PL/SQL functions to meet the requirements of your users. You can use these functions complicated calculations. User-defined PL/SQL functions become part of the total set of PL/SQL functions available to all database processes. To create PL/SQL functions, use SQL*Plus, or a procedural editor. Refer to the *Oracle SQL Language Reference Manual* for more information.

The Oracle Financial Data Manager provides three predefined Exchange Rate Functions/Procedures necessary for multicurrency reporting. They are:

- **The First Function:** Convert_Balance(Balance, From Currency, To Currency, Effective Date)

- **The Second Function:** Convert_Balance_Avg(Balance, From Currency, To Currency, Begin Date, End Date)

- **The Third Function:** Get_Average_Exchange_Rate(From Currency, To Currency, Begin Date, End Date)

These functions, which are described in the following paragraphs, are part of a built-in OFSA database package called OFSA_RATES and can be used for either of these purposes:

- Access exchange rates as of a particular date

- Calculate average exchange rates over a range of dates

4.6.1 The First Function

The function **Convert_Balance(Balance, From Currency, To Currency, Effective Date)** takes as input Balance to be converted, From Currency, To Currency, and Effective Date, and converts the balance based on the exchange rate on the effective date.

4.6.2 The Second Function

The function **Convert_Balance_Avg(Balance, From Currency, To Currency, Begin Date, End Date)** takes as input Balance to be converted, From Currency, To Currency, Begin Date, and End Date, and converts the balance based on the day-weighted average exchange rate between and including the Start Date and End Date.

4.6.3 The Third Function

The function **Get_Average_Exchange_Rate(From Currency, To Currency, Begin Date, End Date)** takes as input From Currency, To Currency, Begin Date, and End Date, and calculates the average exchange rate between the From Currency and To Currency. The average is calculated from all exchange rate instances found between the Begin Date and End Date.

To be accessible to Discoverer software, these PL/SQL functions must be registered in the EUL. Once registered, user-defined PL/SQL functions appear in the list of database functions in the Edit Calculation dialog and can be used for creating or editing calculated items in the same way as the Oracle-supplied functions.

> **Note:** In the User Edition, folders that contain derived items using PL/SQL functions are not visible to users who do not have EXECUTE privileges on those functions. To make those functions accessible, ensure that the user has the associated EXECUTE privilege in the database.

You register PL/SQL functions using the import method of the PL/SQL Functions dialog in the Oracle Discoverer Administration Edition. Importing functions brings in all of the relevant information about each function, including name, database link, return type, and the list of arguments. Importing ensures correct information about the function, because the information does not have to be manually entered on a function-by-function basis. For more information, see the *Oracle Discoverer 3.1 Administration Guide*.

To register the three multi-currency conversion functions, complete the following steps:

1. Login to Discoverer Administration Edition as the EUL owner.

2. Choose Tools | Register PL/SQL Functions.

    The PL/SQL Functions dialog opens.

3. Select the Functions tab and click Import.

    The Import PL/SQL Functions dialog opens, where you can select one or more PL/SQL functions to register.

    Scroll down and select the function to import (such as <database_owner>.OFSA_RATES.CONVERT_BALANCE) and click OK.

    All of the pertinent information about the selected functions is imported, and you do not have to manually enter information or validate it.

4. Click Apply.

5. Repeat steps 3and 4 for the remaining functions:

    ■   <database_owner>.OFSA_RATES.CONVERT_BALANCE_AVG

    ■   <database_owner>.OFSA_RATES.GET_AVERAGE_EXCHANGE_RATE

6. Click OK to close the PL/SQL Functions dialog.

5

FDM/Discoverer Integrator Error Messages

This chapter provides a complete list of all the possible error messages you can encounter in The FDM/Discoverer Integrator application. The following table lists the error messages in alphabetical order. An explanation of what conditions each error message may appear and the action to correct the error is also provided.

Table 5–1   FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
A Folder in the Business Area has missing items.	A Folder in a Business Area is missing items.	Use the Discoverer Administrator Edition to add the missing items, and try again.
All of the columns for table are currently defined within the EUL.	All of the columns for the table are currently defined within the EUL.	Restart the wizard and select another table.
Calculated rate item not created for "<non-weighted rate>". Missing related balance item "<related balance>".	Missing item in EUL: FDM/Discoverer Integrator requires that the weighting balance item for a rate item be defined within the EUL before the rate relationships are generated. The PROPERTY_COLUMN field for the related PROPERTIES table has not been loaded into the EUL.	Load the balance item before loading the rate item.

Table 5–1   (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
Calculated rate item not created for "\<weighted rate>". Missing related balance column "\<related balance>".	Missing column in source table: FDM/Discoverer Integrator requires that the weighting balance item for a rate item be defined within the EUL before the rate relationships are generated. The PROPERTY_COLUMN field for the related field in the OFSA_ COLUMN_PROPERTIES is not a column in the table containing the rate column.	Rerun the transformation process which created the table.
Cannot create Hierarchy Filter. An Item is missing from the Folder.	System cannot create the Hierarchy Filter.	Use The Discoverer Administration Edition to add the missing Item and try again.
Cannot create relationships because the OFSA System Business Area is missing.	System failed to create relationships for an item.	Regenerate the Template Folder by logging out and reconnecting to the database.
Cannot create the OFSA System Business Area. It probably already exists, but you may not have permission to see it.	User tried to create an OFSA System Template Folder which already exists, or is restricted from the user.	Regenerate the Template Folder by logging out and reconnecting to the database.
Cannot create the OFSA System Folders. They may exist in another Business Area. Please remove them.	User tried to create an OFSA System Template Folder which already exists.	Use The Discoverer Administration Edition to remove the duplicate OFSA System Template Folder.
Cannot delete EUL.	User attempted to delete the EUL.	This operation is not allowed.
Cannot find a valid EUL Business Area. Unable to generate folders.	User tried to generate folders when the EUL Business Area was invalid or nonexistent.	Consult your system administrator or Oracle Support Representative.
Cannot find the column name in the table.	System cannot find the given column name in the table.	Use The Discoverer Administration Edition to add the missing Column and try again.
Cannot find the folder in the OFSA System Business Area.	A required Folder in the OFSA System Template is missing.	Regenerate the Template Folder by logging out and reconnecting to the database.

Table 5–1 (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
Cannot move folder to a Business Area which is already its parent.	User tried to move a Folder to a Business Area which already contains the same Folder.	This operation is not allowed.
Cannot open Logfile.	System tried to open a log file on the server.	Consult your system administrator or Oracle Support Representative.
Cannot read template tables meta data for individual code value descriptions.	FDM/Discoverer Integrator failed to access the OFSA System template table for individual code value descriptions. The Template folder is either inaccessible or is invalid.	Regenerate the template folder by logging out and reconnecting to the database.
Cannot read template tables meta data for OFSA_RESULT_SCENARIO.	FDM/Discoverer Integrator failed to access the OFSA System template table for OFSA_RESULT_SCENARIO. The Template folder is either inaccessible or is invalid.	Regenerate the template folder by logging out and reconnecting to the database.
Cannot read template tables meta data for OFSA_RESULT_TYPE_DSC.	FDM/Discoverer Integrator failed to access the OFSA System template table for OFSA_RESULT_TYPE_DSC. The Template folder is either inaccessible or is invalid.	Regenerate the template folder by logging out and reconnecting to the database.
Cannot set Folder into filter. Folder has already been set.	A Folder has been set into a filter twice.	This operation is not allowed.
Cannot share a Folder with a Business Area which is already its parent	User tried to share a Folder with a Business Area which already contains the same Folder.	This operation is not allowed.
Column cannot be found.	A required column was not found.	Consult your system administrator or Oracle Support Representative.
End User Layer Validation Failure. Cyclic Relationship.	EUL object has cyclic relationship.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.

Table 5–1   (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
End User Layer Validation Failure. Exclusivity Failure.	EUL object has exclusivity failure.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
End User Layer Validation Failure. The attribute was invalid:	EUL object has invalid attribute.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
End User Layer Validation Failure. The mandatory attribute was null.	EUL object has its mandatory attribute set to null.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
End User Layer Validation Failure. The mandatory relationship was null.	EUL object has its mandatory relationship set to null.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
End User Layer Validation Failure. The name must be unique.	EUL object has non-unique name. This error occurs when two columns have identical display names within the same table under two conditions: ■ table contains bad metadata where display names are not unique, or ■ FDM/Discoverer Integrator tries to generate a balance-weighted rate column with conflicting names within the same table.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.

Table 5–1   (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
End User Layer Validation Failure. The relationship was invalid.	EUL object has invalid relationship.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
End User Layer Validation Failure. Unknown cause.	EUL object encountered an unknown error.	Consult your system administrator or Oracle Support Representative.
Error Loading datasource %S	Log on connection to database failed. Database is not accessible.	Consult your system administrator or Oracle Support Representative.
Error message cannot be found.	Error message files cannot be found. Files are not accessible, or are invalid.	Consult your system administrator or Oracle Support Representative.
Error while connecting to EUL.	Integrator failed to connect to the Discoverer EUL, which is either not accessible or is invalid.	Consult your system administrator or Oracle Support Representative.
Failed to create relationship items.	System failed to create relationships for an item.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
Failed to unwrap object.	System failed to unwrap an object.	Consult your system administrator or Oracle Support Representative.
Internal Adapter Error. Cannot add item to array.	System adapter failed to add item to array.	Consult your system administrator or Oracle Support Representative.
Internal Adapter Error. Cannot remove item from array.	System adapter failed to remove item from array.	Consult your system administrator or Oracle Support Representative.
No columns were found to load from table	There are no columns to load from the table.	Restart the wizard and select another table.

Table 5–1   (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
Outer Join attributes cannot be changed.	User tried to change the Outer Join attributes.	This operation is not allowed.
Table cannot be found.	A table cannot be found.	Use Discoverer Administration Edition to add the missing table.
Table is being loaded twice.	A table has been loaded twice.	This operation is not allowed.
The Business Area must be named.	Business Area is not named.	Name the Business Area and try again.
The column name cannot be changed.	User tried to rename a restricted column.	This operation is not allowed.
The copied Folder is missing Columns.	A copied Folder is missing Columns.	This operation is not allowed.
The folder must be named.	Folder is not named properly.	Name the folder and try again.
The following Business Area is missing Folders.	A Business Area is missing Folders.	Use The Discoverer Administration Edition to add the missing folders and try again.
The formula cannot be changed.	User tried to change a restricted formula.	This operation is not allowed.
The item must be named.	An item in not named properly.	Name the item and try again.
The Join must be named.	A Join is not named properly.	Name the Join and try again.
The Master Table cannot be changed.	User tried to change the Master Table name.	This operation is not allowed.
The OFSA System Business Area has been changed. Please remove it or correct the problem.	The System has detected that the OFSA System Template Folder has been changed.	Use The Discoverer Administration Edition to remove or correct the OFSA System Template Folder.
The table name cannot be changed.	User tried to rename a restricted table.	This operation is not allowed.

Table 5–1   (Cont.) FDM/Discoverer Integrator Error Messages

Error Message	Condition	Action
The underlying object has changed in the End User Layer. Please reconnect to see changes.	User tried to select an EUL object that has been changed by another user.	Log out and reconnect to the database. If error persists, consult your system administrator or Oracle Support Representative.
The underlying object no longer exists in the End User Layer.	User tried to select an EUL object that no longer exists.	Log out and reconnect to the database. If error persists, consult your system administrator or Oracle Support Representative.
There are no folders to load.	No folders to load; possible data error.	Consult your system administrator who should run the check utility SQL script provided with the package. Once the data problem is identified, correct the data and try again.
This Business Area is required by OFSA. Do not change it in any way.	This is a warning message in the description property of the OFSA System Template Folder.	This operation is not allowed.
User is missing from Root. Cannot create Template Business Area.	User name is missing from the Root.	Regenerate the Template Folder by logging out and reconnecting to the database.
You do not have access to OFSA system data.	A user does not have sufficient access to update or insert into any OFSA tables. A Business Area User is typically a Reporting Data Mart User. Through the OFDM_R_REPORT_MART role, the Reporting Data Mart User is granted the SELECT privilege to the tables or views it requires for running FDM/Discoverer Integrator.	Consult your system administrator or Oracle Support Representative.

Part II

Standard Reports for Oracle Discoverer

The Standard Reports for Oracle Discoverer provide a basic report package based on the FDM database. Part II assumes that:

- Oracle Discoverer is installed

- The OFSA_EULOWNER EUL exists

- The Business Areas have been installed

- The .DIS files are installed on the PC of each user running the reports

- You have transformed your data using the Transformation ID

- You have completed the setup of the Business Areas with FDM/Discoverer Integrator

For information on the installation of these items, refer to either the *Oracle Discoverer Release 3.1 Installation Guide*, or to Chapter 13, "Installing and Configuring Discoverer," in the *Oracle Financial Services Installation and Configuration Guide*.

For information on using the Transformation ID, see the "Transformation ID" chapter in the *Oracle Financial Data Manager Balance & Control Reference Guide*.

This part contains the following chapters:

- Chapter 6, "Overview of the Standard Reports for Oracle Discoverer"

- Chapter 7, "Oracle Budgeting & Planning Reports"

- Chapter 8, "Oracle Customer Householding Reports"

- Chapter 9, "Oracle FDM Administration Reports"

- Chapter 10, "Oracle FDM Common Reports"

- Chapter 11, "Oracle FDM System Reports"

6

Overview of the Standard Reports for Oracle Discoverer

The FDM/Discoverer Integrator application integrates the Oracle Financial Services (OFS) applications and Oracle Discoverer, giving users the ability to perform queries on the standard FDM database. This ability enables the user to develop instant workbooks on a wide variety of information.

Standard reports are part of your installation of the OFS applications and Discoverer. You can implement these reports as is, or modify them to the specifications of your users. Within minutes, you can get valuable information such as a balance sheet for quick decision-making.

This chapter describes the advantages of Discoverer and how to set up the Standard Reports. For more in-depth coverage on the Discoverer tool, refer to the *Oracle Discoverer 3.1 Administration Guide*.

6.1 Advantages of Discoverer

Discoverer is a data query and analytical reporting tool that provides the ability to create ad hoc queries and reports directly on the FDM database. It enables users at all levels of the organization to gain immediate access to important financial data.

Discoverer has many features that provide flexible and in-depth querying and reporting. The key elements are:

- Tabular and cross-tab reporting
- Drill down and drill up capability
- Drill through to detail
- Batch scheduling

- Business Charting

- Web Publishing

Discoverer uses an intuitive "workbook/worksheet" model for saving user queries. Each workbook contains one or more worksheets, and each worksheet represents a query or report. Therefore, each Standard Report is a worksheet within a workbook.

6.2 Accessing the Standard Reports

You can view the reports in the Universal Demo Database at any time. A predefined EUL called OFSA_STANDARD_REPORTS exists for this very purpose. This EUL is an ideal practice area for viewing the reports and testing modifications.

However, to use the reports in the OFSA_EULOWNER EUL against your database, you must load the reports (.dis files) onto the PC and map them to a Business Area. For more information on installing the EUL and the report files, see Chapter 13, "Installing and Configuring Discoverer" in the *Oracle Financial Services Installation and Configuration Guide.*

6.2.1 Preparing for Report Generation

The following steps provide guidelines for preparing for report generation:

1. Open the workbooks on the PC.

2. Determine which reports you want to use.

    Refer to the remaining chapters in this book part for detailed information on the reports.

3. Determine which of your own tables to use for each of the table types required for your reports.

You are now ready to customize the Standard Reports to meet your reporting needs.

6.2.2 Customizing the Reports

Most of the Standard Reports require some manipulation before you can use them. Within each report, Discoverer records the database name, name of the Business Area, and folder names. Since the Standard Reports are prebuilt, this information is not suitable for your database. Therefore, the first time you access the Standard Reports, Discoverer prompts you for the new mapping information.

The following steps assist you in the remapping process:

1. Create a new Business Area that contains the necessary folders for the reports using either FDM/Discoverer Integrator or Discoverer Administration.

2. Register the necessary folders in the new Business Area using FDM/Discoverer Integrator.

   When you open the Standard Reports in the OFSA_EULOWNER EUL against your database, Discoverer starts the mapping process for the reports. Since the Business Area and folders are named differently, a prompt appears to specify the source of the workbook.

3. Choose "Open the Workbook in the Current Database Account" and click OK to start the mapping process.

   The Missing Item or Condition dialog box appears.

4. Select Substitute to bring up the Substitute Item box.

5. Choose the Business Area that contains the appropriate folders and expand the folder that contains the identified item.

6. Locate the item and press OK.

   The screen returns to the original substitute box enabling you to continue mapping all of the items until finished. Once mapping is complete for all items, a prompt appears to run the report.

7. Click Yes to run the report, and save it to either the database or the hard drive.

During this process, you can use different items or folders in the Standard Reports by mapping to a different folder and column. If you do not want one of the items, select Ignore.

This mapping process enables you to modify the Standard Reports to your specifications, with very little extra work. Some standard areas for substitution are hierarchies, rate and balance columns, and date fields.

6.2.3 Replacing the Company Logo

You can replace the Oracle Financial Services bitmap with your company logo by placing the company.bmp file in the C:\ directory. Discoverer searches the root directory for the .bmp file each time you access one of the Standard Reports.

6.2.4 List of Reports

The following are lists of the Standard Reports and their Discoverer file names. You can find descriptions and technical detail on each report in the remaining chapters of Part II.

Note: Standard Reports that are customizable through substitution in the Discoverer End User Edition are marked with an asterisk (*). You must determine which of your own tables to use for each of the table types required for the reports.

Budgeting & Planning Reports

Combined Balance Sheet/Income Statement (bpcomb.dis) *

Combined Balance Sheet/Income Statement Multicurrency (bpcomb_mc.dis) *

Customer Householding Reports

Account Reports (chacct.dis)

Age Reports (chcuage.dis)

Branch Report (chcubran.dis)

Cross Sell Report (chxsell.dis)

Customer Reports (chcust.dis)

Gender Reports (cucugend.dis)

Household Reports (chhsehld.dis)

FDM Administration Reports

Database Object Privileges Report (dobjpriv.dis)

Description Table Mappings Report (descmap.dis)

Leaf Objects Report (leaftab.dis)

Object Management Report (objects.dis)

Security Profile Privileges Report (secuprof.dis)

User Group Privileges Report (ugrppriv.dis)

User Privileges Report (userpriv.dis)

FDM Common Report

The FDM common report is Hierarchy Report for Organizational Unit (hierarc.dis) *

FDM System Reports

Error Message Report (errormsg.dis)

Ledger Stat Instrument Report (lsinstr.dis)

Ledger Stat Reconciliation Report (recon.dis)

Performance Analyzer Reports

Allocation Description Report (allocate.dis)

Audit Trail Report (audit.dis)

Balance Sheet (balance.dis) *

Balance Sheet Multicurrency (balance_mc.dis) *

Balance Sheet and Income Statement (inc-bal.dis) *

Balance Sheet and Income Statement Multicurrency (inc-bal_mc.dis) *

Income Statement (income.dis) *

Income Statement Multicurrency (income_mc.dis) *

Instrument Transaction Drill-to-Detail Report (instrans.dis) *

Ledger Stat Drill-to-Detail Report (lidrill.dis) *

Single Stratification Report (strat1.dis) *

Single Stratification Multicurrency (strat1_mc.dis) *

Single Stratification User Description Report (strat1d.dis) *

Double Stratification Report (strat2.dis) *

Double Stratification Multicurrency (strat2_mc.dis) *

Double Stratification User Description Report (strat2d.dis) *

Double Stratification User Description Multicurrency (strat2d_mc.dis) *

Rate Manager Reports

The Rate Manager report is Exchange Rate Report (exchng.dis)

Risk Manager Reports

Detail Cashflow Runoff Report (detcfrun.dis)

Earning at Risk Density Report (eardensi.dis) *

Earning at Risk Detail Report (eardtl.dis) *

Earning at Risk Summary Report (earsum.dis) *

Income vs. Rate Report (inc-rate.dis) *

Detail/Summary Cashflow Report (rmcashflw.dis) *

Consolidated Detail/Summary Cashflow Report (rmcashflw_mc.dis) *

GAP Detail Report (rmgapdet.dis) *

Consolidated GAP Detail Report (rmgapdet_mc.dis) *

GAP Summary Report (rmgapsum.dis) *

Consolidated GAP Summary Report (rmgapsum_mc.dis) *

Scenario Income Statement (rmincsce.dis) *

Consolidated Scenario Income Statement (rmincsce_mc.dis) *

Summary Income Statement (rmincsum.dis) *

Consolidated Summary Income Statement (rmincsum_mc.dis) *

Market Value Report (rmmktval.dis) *

Product, Rank, VaR Report rmvar1.dis) *

Rank, VaR (Total Bank Level) Report (rmvar2.dis) *

Rank, Product, VaR Report (rmvar3.dis) *

Stochastic Market Value Report (stochmkt.dis) *

Stochastic Rates Report (stochrates.dis)

Term Structure Parameters Report (termspar.dis)

FASB 133 Report (fasb133.dis) *

Transfer Pricing Reports

Transfer Pricing Income Statement (rminctp.dis) *

Transfer Pricing Option Cost Instrument Report (tpocinstr.dis) *

7

Oracle Budgeting & Planning Reports

The Oracle Budgeting & Planning reports address typical business requirements for line managers and supervisors. The reports provide reviews of financial results in a forecasting or budgeting exercises.

This chapter discusses the Budget & Planning balance sheet reports, including the following:

- Combined Balance Sheet/Income Statement

- Drillable Balance Sheet

- Drillable Income Statement

- Spread Income Statement

This chapter also explains the Combined Balance Sheet/Income Statement Multicurrency report.

7.1 Budgeting & Planning Balance Sheet Reports

This workbook contains four reports. Each report represents a different view of the balance sheet and income statement.

The workbook name is **bpcomb.dis**.

7.1.1 Combined Balance Sheet/Income Statement

This report shows the combined balance sheet and income statement to be used for Budgeting & Planning.

Folder

OFS Rpt Ledger Stat

Items

Ofs Rpt Ledger Stat.End Date

Ofs Rpt Ledger Stat.Charge/Credit

Calculations

Table 7–1   Combined Balance Sheet/Income Statement Report Calculations

Item Name	Formula	Description
Balance Sheet (in millions)	SUM(NULL)	Label
Income Statement (in thousands)	SUM(NULL)	Label
Interest Expense	SUM(DECODE(Category Description,'Total Liabilities & Capital',Interest,0))/1000	Returns the total Interest expense
Interest Income	SUM(DECODE(Category Description,'Total Assets',Interest,0))/1000	Returns the total Interest income
Net Income	SUM(DECODE(Category Description,'Total Assets',Interest,0))/1000-SUM(DECODE(Category Description,'Total Liabilities & Capital',Interest,0))/1000+SUM(DECODE(Category Description,'Non-Interest Income',Non Interest Income,0))/1000-SUM(DECODE(Category Description,'Non-Interest Expense',Non Interest Expense,0))/1000	Returns net income by adding interest income with non-interest income and subtracting interest expense and non-interest expense.
Net Margin	SUM(DECODE(Category Description,'Total Assets',Interest,0))/1000-SUM(DECODE(Category Description,'Total Liabilities & Capital',Interest,0))/1000	Returns the difference between Interest Income and Interest Expense.

Table 7–1   (Cont.) Combined Balance Sheet/Income Statement Report Calculations

Item Name	Formula	Description
Net Non-Interest Income	SUM(DECODE(Category Description,'Non-Interest Income',Non Interest Income,0))/1000-SUM(DECODE(Category Description,'Non-Interest Expense',Non Interest Expense,0))/1000	Returns the difference between Non Interest Income and Non Interest Expense.
Non-Earning Assets	SUM(DECODE("Sub-Category Description",'Total Nonearning Assets',Average Bal,0))/1000000	Returns the Average balance for Non Earning Assets and divides by 1 million.
Non-Interest Expense	SUM(DECODE(Category Description,'Non-Interest Expense',Non Interest Expense,0))/1000	Returns the value from Non Interest Expense.
Non-Interest Income	SUM(DECODE(Category Description,'Non-Interest Income',Non Interest Income,0))/1000	Returns the value from Non Interest Income.
Total Assets	SUM(DECODE(Category Description,'Total Assets',Average Bal,0))/1000000	Returns the average balance for Assets and divides by 1 million.
Total Earning Assets	SUM(DECODE("Sub-Category Description",'Total Earning Assets',Average Bal,0))/1000000	Returns the Average balance for Earning Assets and divides by 1 million.
Total Liabilities & Capital	SUM(DECODE(Category Description,'Total Liabilities & Capital',Average Bal,0))/1000000	Returns the average balance for Liabilities and Capital and divides by 1 million.

Filter

Consolidation Code = 200

Join

BP_COA_ROLLUP: PRODUCT HIER (COMMON_COA_ID)
> Ofs Rpt Ledger Stat

7.1.2 Drillable Balance Sheet

This report shows the drillable balance sheet to be used for budgeting and planning.

Folders

Ofs Rpt Ledger Stat

BP_COA_ROLLUP: PRODUCT HIER

Items

Ofs Rpt Ledger Stat.End Date

BP_COA_ROLLUP: PRODUCT HIER.Category Description

BP_COA_ROLLUP: PRODUCT HIER.Sub-Category Description

Calculations

Item Name	Formula	Description
Average Balance	SUM(Average Bal)∕1000000	Returns the average balance and divides by 1 million.

Filters

Category Description IN ('Total Liabilities & Capital','Total Assets')

Category Description = 'Total Assets'

Category Description = 'Total Liabilities & Capital'

Consolidation Code = 200

Join

BP_COA_ROLLUP: PRODUCT HIER (COMMON_COA_ID)
> Ofs Rpt Ledger Stat

7.1.3 Drillable Income Statement

This report shows the drillable income statement to be used for budgeting and planning.

Folders

Ofs Rpt Ledger Stat

RPT_PROD_HIER: RPT_PROD_HIER

Items

RPT_PROD_HIER: RPT_PROD_HIER.Category Description

RPT_PROD_HIER: RPT_PROD_HIER.Sub-Category Description

Ofs Rpt Ledger Stat.End Date

Calculations

Item Name	Formula	Description
Interest	SUM(DECODE("RPT_PROD_HIER: RPT_PROD_HIER". Category Description,'Total Interest Income',Interest,0))∕ 1000+SUM(DECODE("RPT_PROD_HIER: RPT_PROD_ HIER".Category Description,'Total Interest Expense', Interest,0))∕1000+SUM(DECODE("RPT_PROD_HIER: RPT_ PROD_HIER".Category Description,'Non-Interest Income', Non Interest Income,0))∕1000+SUM(DECODE("RPT_PROD_ HIER: RPT_PROD_HIER".Category Description, 'Non-Interest Expense',Non Interest Expense,0))∕1000	Returns the net income.

Filters

Category Description IN ('Non-Interest Expense','Non-Interest Income','Total Interest Expense','Total

Category Description = 'Total Interest Income'

Consolidation Code = 200

Join

RPT_PROD_HIER: RPT_PROD_HIER1 > Ofs Rpt Ledger Stat 2

7.1.4 Spread Income Statement

This report shows the spread income statement to be used for budgeting and planning.

Folders

Ofs Rpt Ledger Stat

RPT_PROD_HIER: RPT_PROD_HIER

Items

RPT_PROD_HIER: RPT_PROD_HIER.Category Description

RPT_PROD_HIER: RPT_PROD_HIER.Sub-Category Description

Ofs Rpt Ledger Stat.End Date

Calculations

Item Name	Formula	Description
Spread	SUM(DECODE("RPT_PROD_HIER: RPT_PROD_ HIER".Category Description,'Total Interest Expense',"Charge/Credit"-Ofs Rpt Ledger Stat.Interest,'Total Interest Income',Ofs Rpt Ledger Stat .Interest-"Charge/Credit",0))/1000	Returns the spread between interest and charge/credit.

Filters

Category Description IN ('Total Interest Expense','Total Interest Income')

Category Description = 'Total Interest Income'1

Category Description = 'Total Interest Expense'1

Consolidation Code = 200

Join

RPT_PROD_HIER: RPT_PROD_HIER1 > OFS Rpt Ledger Stat 2

7.2 Combined Balance Sheet/Income Statement Multicurrency

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency specified by the user. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features" for more information.

The workbook name is bpcomb_mc.dis.

8

Oracle Customer Householding Reports

Integration between Oracle Customer Householding (CH) and Oracle Discoverer provides dynamic and flexible reporting for business and marketing professionals.

When you log into the Customer Householding database as OFSA_EULOWNER, a predefined Business Area (called CH) appears. Through this Business Area, you have immediate access to your data without having to invest time in learning the Customer Householding table structure. This structure also includes some predefined filters that you can use in your queries.

In addition, several standard reports are part of the installation. Each of these reports is a useful demonstration of the types of reports that can be generated by Discoverer.

Discoverer standard reports are included in the implementation of Customer Householding. The standard reports are worksheets categorized and grouped into workbooks.

These reports exploit a wide range of the capabilities of Discoverer. The following list provides the name and a brief description of each report.

8.1 Account Report

The Account Report worksheets are as follows:

Worksheet Name	Description
Balance Stratification	Summary of all accounts by balance stratification range - stratification range populated by Oracle Balance and Control and stored in the database.
Branch & Product Summary	Summary of all accounts by Product and Branch; Branch name listed in page detail section of report.
Closed Accounts	Accounts closed by Product type for a range of dates specified by user input from parameter function of report.
Closed Accounts by Branch	Accounts closed by Product type and Branch (Branch in page detail section) for a range of dates specified by user input from parameter function of report.
Open Accounts	Accounts opened by Product type for a range of dates specified by user input from parameter function of report.
Open Accounts by Branch	Accounts opened by Product type and Branch (Branch in page detail section) for a range of dates specified by user input from parameter function of report.
Product Detail	Income statement details in dollars, basis points and average account at the Subproduct level.
Product Hierarchy Drilldown	Product summary using hierarchy drilldown feature of Discoverer.
Product Summary	Summary of all accounts by Product.
Product Summary in Basis Points	Product summary displaying average account information in basis points of average balance.
Product Summary with Averages	Product summary displaying average account information in dollars.
Service Summary	Summary of all accounts by Service.
Subproduct Summary	Summary of all accounts by Subproduct - Service Type and Service are listed in page detail section.

8.2 Age Report

The Age Report worksheets are as follows:

Worksheet Name	Description
Age < 20	Shows the summary of all customers of age < 20.
Age > 60	Shows the summary of all customers of age > 60.
Age between 20 & 30	Shows the summary of all customers of age between 20 & 30.
Age between 30 & 40	Shows the summary of all customers of age between 30 & 40.
Age between 40 & 50	Shows the summary of all customers of age between 40 & 50.
Age between 50 & 60	Shows the summary of all customers of age between 50 & 60.

8.3 Branch Report

The Brance Report contains the Branch Prompt worksheet, which shows the customer data segmented by branch for customer householding.

8.4 Cross Sell Report

The Cross Sell Report contains the Cross Sell worksheet. This worksheet shows all possible combinations of services (services listed down the side and across the top of the report); product balance and number of accounts information are included.

8.5 Customer Report

The Customer Report worksheets are as follows:

Worksheet Name	Description
3-digit Zip Code	Summary of customers by 3-digit zip code.
Age Segment	Summary of customers by Age Segment - Age Segment populated using Oracle Balance and Control and stored in the database.
Branch Name	Summary of customers by Branch Name.
Customer Account Detail	Customer account detail (product, account number, balance, and contribution) combined with summary customer information - Branch name, contribution decile and customer names in page detail section - customer names appear in pick list.
Customer Detail	Customer details grouped by officer and contribution decile (in page detail section) - in descending contribution order.
Decile Segment	Summary of customers by contribution decile.
Decile Segment Averages	Summary of customers by contribution decile - by average account in dollars.
Officer	Summary of customers by officer name.
Retention and Decile	Summary of customers by retention and decile segments.
ROE & Decile Segments	Summary of customers ROE and Decile segments.
ROE Segment	Summary of customers by ROE segment.
Years with Organization	Summary of customers grouped by number of years with the organization.

8.6 Gender Report

Worksheet Name	Description
Female	Shows the customer data segmented by female gender.
Male	Shows the customer data segmented by male gender.

8.7 Household Report

The Household Report contains the following worksheets:

Worksheet Name	Description
3-digit Zip Code	Summary of households by 3-digit zip code.
Branch Name	Summary of households by branch name.
Branch Number	Summary of households by branch number.
Contribution Decile	Summary of households by household contribution decile segment.
Household Customer Detail	Household summary combined with customer detail; user inputs parameters for branch number and household contribution decile segment; households appear in drop down pick list.
Household Detail	Household detail grouped by officer and household contribution segments (in page detail section).
Officer	Summary of households by officer name.

9

Oracle FDM Administration Reports

The FDM Administration reports provide access to administrative information including management reports on description mappings, table classifications, object privileges, and object and user registration.

This chapter provides details on the following reports:

9.1 Description Table Mappings Reports

The reports for Description Table Mappings are available in the following workbook:

The workbook name is descmap.dis.

This workbook contains the following reports:

- Description Mappings by Table Report
- Description Mappings by Column Report

9.1.1 Description Mappings by Table Report

This report shows the Code Description mappings by table for FDM registered tables or views.

Folder

FDMA Description Table Mappings

Items

FDMA Description Table Mappings.Table Name

FDMA Description Table Mappings.Column Name

FDMA Description Table Mappings.Description Join Column Name

Sort On

FDMA Description Table Mappings.Column Name

9.1.2 Description Mappings by Column Report

This report shows the Code Description mappings by column for FDM registered tables or views.

Folder

FDMA Description Table Mappings

Items

FDMA Description Table Mappings.Table Name

FDMA Description Table Mappings.Column Name

FDMA Description Table Mappings.Description Join Column Name

Sort On
FDMA Description Table Mappings.Table Name

9.2 Database Object Privileges Reports

The reports for database object privileges are available in the following workbook:

The workbook name is dobjpriv.dis.

This workbook contains the following reports:

- Privileges by Role Report

- Privileges by Object Report

9.2.1 Privileges by Role Report

This report shows privileges by role on FDM registered objects by the Recipient Name (role name) or the Object Name.

Folder
FDMA Database Object Privilege Assignments

Items
FDMA Database Object Privilege Assignments.Recipient Name

FDMA Database Object Privilege Assignments.Object Name

FDMA Database Object Privilege Assignments.Db Object Privilege

FDMA Database Object Privilege Assignments.Protected Flg

Sort On
FDMA Database Object Privilege Assignments.Object Name

9.2.2 Privileges by Object Report

This report shows privileges by object on FDM registered objects by the Recipient Name (role name) or the Object Name.

Folder

FDMA Database Object Privilege Assignments

Items

FDMA Database Object Privilege Assignments.Recipient Name

FDMA Database Object Privilege Assignments.Object Name

FDMA Database Object Privilege Assignments.Db Object Privilege

FDMA Database Object Privilege Assignments.Protected Flg

Sort On

FDMA Database Object Privilege Assignments.Recipient Name

9.3 Leaf Objects Reports

The reports for Leaves Registered Objects are available in the following workbook:

The workbook name is leaftab.dis.

This workbook contains the following reports:

- Registered Objects for 'All' Leaves Report
- Registered Objects for 'Ledger' Leaves Report

9.3.1 Registered Objects for 'All' Leaves Report

This report shows the tables and views which must be updated in order for a new Leaf Column of type "All" to be registered for the FDM metadata.

The characteristics are as follows:

- Worksheet: Objects Reg. for 'All' Leaves.
- Folder: FDMA Table Classification Assignments
- Item: FDMA Table Classification Assignments.Table Name
- Sort on: FDMA Table Classification Assignments.Table Name

- Filter: Table Classification Cd IN (50,200,210,300,310,350,360,370)

9.3.2 Registered Objects for 'Ledger' Leaves Report

This report shows the tables and views which must be updated in order for a new Leaf Column of type "Ledger Only" to be registered for the FDM metadata.

The characteristics are as follows:

- Worksheet name: Objects Reg. for 'Ledger' Leaves
- Folder: FDMA Table Classification Assignments
- Item: FDMA Table Classification Assignments.Table Name
- Sort on: FDMA Table Classification Assignments.Table Name
- Filter: Table Classification Cd = 352

9.4 Objects Management Reports

The objects reports are available in the following workbook:

The workbook name is objects.dis.

This workbook contains the following reports:

- Registered Objects Report
- Registered Columns Report
- Column Properties Report
- Table Classification Assignments Report

9.4.1 Registered Objects Report

This report shows the registered objects information about tables and views registered in the Financial Data Manager (FDM) metadata.

Folder
FDMA Tables

Items

FDMA Tables.Table Name

FDMA Tables.Dbf Name

FDMA Tables.Display Name

FDMA Tables.Description

Sort On

FDMA Tables.Table Name

9.4.2 Registered Columns Report

This report shows the registered columns information about tables and views registered in the Financial Data Manager (FDM) metadata.

Folders

FDMA Data Type Descriptions

FDMA Tab Columns

Items

FDMA Data Type Descriptions.Ofsa Data Type

FDMA Tab Columns.Table Name

FDMA Tab Columns.Column Name

FDMA Tab Columns.Dbf Name

FDMA Tab Columns.Display Name

FDMA Tab Columns.Description

Sort On

FDMA Tab Columns.Column Name

Join

FDM Tab Columns > FDM Data Type Descriptions

9.4.3 Column Properties Report

This report shows the column properties information about tables and views registered in the Financial Data Manager (FDM) metadata.

Folders

FDMA Column Property Assignments

FDMA Column Property Descriptions

Items

FDMA Column Property Assignments.Table Name

FDMA Column Property Assignments.Column Name

FDMA Column Property Assignments.Related Field

FDMA Column Property Descriptions.Column Property

Sort On

FDMA Column Property Assignments.Column Name

FDMA Column Property Descriptions.Column Property

Join

Column Property Assignments > Column Property Descriptions

9.4.4 Table Classification Assignments Report

This report shows the table classification assignments information about tables and views registered in the Financial Data Manager (FDM) metadata.

Folders

FDMA Table Classification Assignments

FDMA Table Classification Descriptions

Items

FDMA Table Classification Assignments.Table Name

FDMA Table Classification Assignments.Table Classification Cd

FDMA Table Classification Descriptions.Table Classification

Sort On

FDMA Table Classification Assignments.Table Classification Cd

Joins

Table Classification Assignments > Table Classification Descriptions

9.5 Security Profile Privileges Reports

The reports for security profiles are available in the following workbook:

The workbook name is secuprof.dis.

This workbook contains the following reports:

- Security Profile Application Privileges Report
- Security Profile Function Privileges Report

9.5.1 Security Profile Application Privileges Report

This report shows the application privileges assigned to FDM Security Profiles.

Folder

FDMA Security Profile Application Privileges

Items

FDMA Security Profile Application Privileges.Application Privilege

FDMA Security Profile Application Privileges.Privilege Source

FDMA Security Profile Application Privileges.Security Profile Name

FDMA Security Profile Application Privileges.Privilege Source Entity Type

Sort On

FDMA Security Profile Application Privileges.Application Privilege

9.5.2 Security Profile Function Privileges Report

This report shows the function privileges assigned to FDM Security Profiles.

Folder

FDMA Security Profile Function Privileges

Items

FDMA Security Profile Function Privileges.Action Name

FDMA Security Profile Function Privileges.Construct Name

FDMA Security Profile Function Privileges.Privilege Source

FDMA Security Profile Function Privileges.Security Profile Name

FDMA Security Profile Function Privileges.Privilege Source Entity Type

Sort On

FDMA Security Profile Function Privileges.Construct Name

9.6 User Group Privileges Reports

The reports for user group privileges are available in the following workbook:

The workbook name is ugrppriv.dis.

This workbook contains the following reports:

- User Group Application Privileges Report
- User Group Function Privileges Report
- User Group Folder Privileges Report
- User Group Dynamic Table Classification Privileges Report

9.6.1 User Group Application Privileges Report

This report shows the application privileges assigned to FDM user groups.

Folder

FDMA User Group Application Privileges

Items

FDMA User Group Application Privileges.User Group Name

FDMA User Group Application Privileges.Application Name

FDMA User Group Application Privileges.Privilege Source

FDMA User Group Application Privileges.Privilege Source Entity Type

Sort On

FDMA User Group Application Privileges.Application Name

9.6.2 User Group Function Privileges Report

This report shows the function privileges assigned to FDM user groups.

Folder

FDMA User Group Function Privileges

Items

FDMA User Group Function Privileges.User Group Name

FDMA User Group Function Privileges.Action Name

FDMA User Group Function Privileges.Construct Name

FDMA User Group Function Privileges.Privilege Source

FDMA User Group Function Privileges.Privilege Source Entity Type

Sort On

FDMA User Group Function Privileges.Construct Name

9.6.3 User Group Role Privileges Report

This report shows the database role privileges assigned to FDM user groups.

Folder

FDMA User Group Role Privileges

Items

FDMA User Group Role Privileges.User Group Name

FDMA User Group Role Privileges.Role Name

FDMA User Group Role Privileges.Role Source

FDMA User Group Role Privileges.Role Source Entity Type

Sort On
FDMA User Group Role Privileges.Role Name

9.6.4 User Group Folder Privileges Report

This report shows the folder privileges assigned to FDM user groups.

Folder
FDMA User Group Folder Privileges

Items
FDMA User Group Folder Privileges.User Group Name

FDMA User Group Folder Privileges.Folder Name

FDMA User Group Folder Privileges.Folder Source

FDMA User Group Folder Privileges.Folder Source Entity Type

Sort On
FDMA User Group Folder Privileges.Folder Name

9.6.5 User Group Dynamic Table Classification Privileges Report

This report shows the dynamic table classification privileges assigned to FDM users.

Folder
FDMA User Group Dynamic Table Classification Privileges

Items
FDMA User Group Dynamic Table Classification Privileges.Table Classification

FDMA User Group Dynamic Table Classification Privileges.Db Object Privilege

FDMA User Group Dynamic Table Classification Privileges.Privilege Source

FDMA User Group Dynamic Table Classification Privileges.Privilege Source Entity Type

FDMA User Group Dynamic Table Classification Privileges.User Group Name

Sort On
FDMA User Group Dynamic Table Classification Privileges.Table Classification

9.7 User Privileges Reports

The reports for user privileges are available in the following workbook:

The workbook name is **userpriv.dis**.

This workbook contains the following reports:

- User Application Privileges Report
- User Function Privileges Report
- User Role Privileges Report
- User ID Folder Privileges Report
- User Dynamic Table Classification Privileges Report

9.7.1 User Application Privileges Report

This report shows the application privileges assigned to FDM users.

Folder
FDMA User Application Privileges

Items
FDMA User Application Privileges.User Name

FDMA User Application Privileges.Application Privilege

FDMA User Application Privileges.Privilege Source

FDMA User Application Privileges.Privilege Source Entity Type

9.7.2 User Function Privileges Report

This report shows the function privileges assigned to FDM users.

Folder

FDMA User Function Privileges

Items

FDMA User Function Privileges.User Name

FDMA User Function Privileges.Action Name

FDMA User Function Privileges.Construct Name

FDMA User Function Privileges.Privilege Source

FDMA User Function Privileges.Privilege Source Entity Type

Sort On

FDMA User Function Privileges.Construct Name

FDMA User Function Privileges.Privilege Source

User Prompt

The user prompt is Username. The user enters the username on which to report.

9.7.3 User Role Privileges Report

This report shows the database role privileges assigned to FDM users.

Folder

FDMA User Role Privileges

Items

FDMA User Role Privileges.User Name

FDMA User Role Privileges.Role Name

FDMA User Role Privileges.Role Source

FDMA User Role Privileges.Role Source Entity Type

9.7.4 User ID Folder Privileges Report

This report shows the folder privileges assigned to FDM users.

Folder

FDMA User Folder Privileges

Items

FDMA User Folder Privileges.User Name

FDMA User Folder Privileges.Folder Name

FDMA User Folder Privileges.Folder Source

FDMA User Folder Privileges.Folder Source Entity Type

9.7.5 User Dynamic Table Classification Privileges Report

This report shows the dynamic table classification privileges assigned to FDM users.

Folder

FDMA User Dynamic Table Classification Privileges

Items

FDMA User Dynamic Table Classification Privileges.Username

FDMA User Dynamic Table Classification Privileges.Table Classification

FDMA User Dynamic Table Classification Privileges.Db Object Privilege

FDMA User Dynamic Table Classification Privileges.Privilege Source

FDMA User Dynamic Table Classification Privileges.Privilege Source Entity Type

10

Oracle FDM Common Reports

The Oracle Financial Data Manager (FDM) Common Reports work on the common components of the Reporting Data Mart. Currently there is one report available in this classification: Hierarchy Report for Organizational Unit. This report displays the complete organization structure, and you can modify it to provide your users with a variety of views.

Report Name

Hierarchy Report for Organizational Unit

Workbook Name

hierarc.dis

Folder

Std Rpt Hier-Org Unit Hier

Items

Std Rpt Hier-Org Unit Hier.State Banks Description

Std Rpt Hier-Org Unit Hier.Total Rollup Description

Std Rpt Hier-Org Unit Hier.Org Unit ID Leaf Description

Std Rpt Hier-Org Unit Hier.Channel Description

Std Rpt Hier-Org Unit Hier.Area Description

Std Rpt Hier-Org Unit Hier.City/County Banks Description

Sort On

Std Rpt Hier-Org Unit Hier.Total Rollup Description

Std Rpt Hier-Org Unit Hier.State Banks Description

Std Rpt Hier-Org Unit Hier.City/County Banks Description

Std Rpt Hier-Org Unit Hier.Area Description

Std Rpt Hier-Org Unit Hier.Channel Description

Std Rpt Hier-Org Unit Hier.Org Unit ID Leaf Description

Filter

Name	Formula	Description
Orphan Filter	Channel Value <> '-99000L01'	FDM/Discoverer Integrator generates this orphan filter to eliminate any orphan numbers. You can enable or disable this filter.

Special Considerations

Any hierarchy can be substituted.

The rollup values can also be added to this report.

11

Oracle FDM System Reports

The FDM System Reports work on the processing data mart, require little modification, and do not require Business Area registration through Oracle FDM/Discoverer Integrator. After you install the FDM Business Area, the reports are ready for generation and modification. For directions on installation, refer to the *Oracle Financial Services Installation and Configuration Guide.*

This chapter discusses the following reports:

- Error Messages Report

- Ledger Stat Instrument Upload Report

- Ledger Stat Reconciliation Reports

11.1 Error Messages Report

This report shows the error messages captured during OFSA application processing.

Workbook

errormsg.dis

Folders

FDM Catalog Of IDs

FDM Ofsa Jobs Run

FDM Ofsa Job Status Dsc

FDM Ofsa Message Log

FDM Ofsa Messages

Items

FDM Catalog Of Ids.Id Desc Short

FDM Catalog Of Ids.Folder Name

FDM Ofsa Jobs Run.Job End Timestamp

FDM Ofsa Job Status Dsc.Job Status

FDM Ofsa Message Log.Context Specific Text

FDM Ofsa Message Log.Job Number

FDM Ofsa Message Log.Sequences

FDM Ofsa Message Log.Message Cd

FDM Ofsa Messages.Message Action

FDM Ofsa Messages.Message Text

Calculations

Name	Formula	Description
Application	DECODE(Application Cd,0,'Performance Analyzer',1,'FDM Administration',2,'Risk Manager',3,'Transfer Pricing',4,'Balance and Control',5,'Portfolio Analyzer',6,'Customer Householding',7,'Market Manager',8,'Discover Integrator',9,'Rate Manager','Other')	Returns the OFS application name based on the Application Code field.

Joins

Ofsa Job Status Dsc > Ofsa Jobs Run

Ofsa Message Log > Ofsa Jobs Run

Ofsa Message Log > Ofsa Messages

Catalog Of Ids > Ofsa Jobs Run

11.2 Ledger Stat Instrument Upload Report

This report provides the data to move non-instrument ledger data such as, expense items into an instrument table.

Workbook

lsinstr.dis

Folder

FDM Ledger Stat

Items

FDM Ledger Stat.Common Coa Id

FDM Ledger Stat.Customer Segment ID

FDM Ledger Stat.Channel ID

FDM Ledger Stat.Gl Account Id

FDM Ledger Stat.Iso Currency Cd

FDM Ledger Stat.Org Unit Id

FDM Ledger Stat.Transaction ID

Calculations

Table 11–1   Ledger Stat Instrument Upload Report Calculations

Item Name	Formula	Description
IDENTITY_CODE	User-defined value	A table name, such as OFSA_Ledger_Stat_ Instrument; A Not Null column in the OFSA_ Ledger_Stat_Instrument table that must be populated. Each month, the administrator should change the Identity Code to a unique number for tracking purposes.
INTEREST_RATE_CD	User-defined value	A Not Null column in the OFSA_Ledger_Stat_ Instrument table that must be populated. Choose a number that helps identify the transaction, such as a user-defined number.

Table 11–1   (Cont.) Ledger Stat Instrument Upload Report Calculations

Item Name	Formula	Description
ORIGINATION_DATE	User-defined value	A Not Null column in the OFSA_Ledger_Stat_Instrument table that must be populated. The default strategy is to subtract one month from the month being reconciled.
AS_OF_DATE	User-defined value	A Not Null columns in the OFSA_Ledger_Stat_Instrument table that must be populated. Set the date for the last day of the month being reconciled.
Month 12 SUM	User-defined value	Month_12 bucket; change the bucket monthly. Only one month can be run at a time.

Filters

Consolidation Cd = 100 (actual data)

Financial Elem Id = 100 (end balance)

Table Name = 'N/A' (no instrument data)

Year S = 1999 (limit the year to 1999)

"Month 12" <> 0 (eliminate zeroes)

Join

Detail Gl Account Id > Ledger Stat

11.3 Ledger Stat Instrument Report

The Ledger Stat Instrument Report populates a table in the database on which transfer pricing (TP) can occur. Transfer Pricing does not work on the Ledger_Stat table, so any figures to be transfer priced must be located in an instrument table. This procedure moves the non-instrument data from the Ledger_Stat table to an instrument table that allows for Transfer Pricing.

The Ledger Stat Instrument Report utilizes tables OFSA_Detail_GL_Account_ID and OFSA_Ledger_Stat_Instrument control file, lsinstr.ctl, and CSV file, lsinstr.csv, to run. The tables, formerly created by the user, are now permanently seeded during the database creation or upgrade process.

11.3.1 Indexing the OFSA_Detail_GL_Account_ID Table

The OFSA_DETAIL_GL_ACCOUNT_ID table contains two columns, which are described in the following table:

Column Name	GL ACCOUNT ID	Table Name
COLUMN_TYPE	NUMBER	VARCHAR2
COLUMN_SIZE	14	30

The database administrator needs to manually maintain the indexing of this table for maximum performance. Remember that GL_ACCOUNT_ID must be unique.

11.3.2 Populating the OFSA_Detail_GL_Account_ID Table

To populate this table, insert distinct GL account IDs from each of the instrument tables along with the instrument table name. Also, you must insert any remaining GL accounts that exist in the OFSA_LEAF_DESC table and do not exist in the Instrument tables into the Detail table along with an N/A for the Table_Name. Here is an example:

```
insert into OFSA_Detail_GL_Account_ID(
    select distinct GL_ACCOUNT_ID, 'DEPOSITS'
    from Deposits)

insert into OFSA_Detail_GL_Account_ID(
    select leaf_node, 'N/A'
    from OFSA_Leaf_desc
where Leaf_Num_ID = 2 and leaf_node not in
    (select gl_account_id from OFSA_Detail_GL_Account_ID))
```

Review these GL account numbers and table names. There may be some cases where the GL account number was missing in the instrument table and has been labeled as N/A. A review of these numbers can help reduce future problems. Also, remember if you are adding GL account numbers, be sure to add them to the OFSA_Detail_GL_Account_ID table.

Likewise, the OFSA_Ledger_Stat_Instrument table, formerly created by the user, is now permanently seeded during the database creation/upgrade process. The database administrator needs to manually maintain the indexing of this table for maximum performance.

> **Note:** Record which columns are not null. These columns MUST be included in the lsinstr.ctl file.

11.3.3 Ledger Stat Instrument Workbook

The Ledger Stat Instrument Report workbook is composed of one worksheet, the Upload sheet. Run the upload sheet and save it as an .CSV file. To do so, choose export and then the drop down box to choose "CSV(Comma Delimited)(*.csv)." This becomes the data file that writes back to the OFSA_Ledger_Stat_Instrument table.

Running the Workbook

The procedure for preparing and running the Ledger Stat Instrument workbook is:

1. Open the Ledger Stat Instrument Report workbook, lsinstr.dis.

2. Set the above conditions and calculations in the workbook.

3. Run the Upload worksheet and export the results to a CSV file.

Populating the OFSA_Ledger_Stat_Instrument Table From the lsinstr.csv File

To populate the OFSA_Ledger_Stat_Instrument table, use the following procedure to modify the lsinstr.ctl file, which is located in the OFSA_INSTALL/dbs/<OFSA release> /utilities/reporting directory on the server side, or in the <Oracle Home>\<OFSA release>\disco31\ directory on the client side.

1. Modify the INFILE name to include the location of the CSV file from above.

2. Include any customer-defined leaves in the list of columns at the bottom of the file.

   Remember that all not null columns in the OFSA_Ledger_Stat_ Instrument table must be included. Also, the order must be the same between the control file and the Discoverer table columns that export to the .csv file.

   > **Note:** Id_number sequence (1,1) (This automatically populates this column with a unique number for each row. Id_number must be unique.)

3. Call the SQL Loader and run the process.

This loads the data from the .csv file to the OFSA_Ledger_Stat_Instrument table. The table should now be populated.

11.4 Ledger Stat Reconciliation Reports

These reports provide validation and reconciliation for the Ledger Stat table and the instrument table.

Workbook

recon.dis

Reports

Ledger Stat Reconciliation Audit Report

Ledger Stat Reconciliation Upload Report

11.4.1 Ledger Stat Reconciliation Audit Report

This report shows all outages between the Ledger Stat table and the instrument table and provides a mechanism to plug data if required.

Folder

FDM Recon SQL

Items

FDM Recon SQL.Common Coa Id

FDM Recon SQL.Customer Segment ID

FDM Recon SQL.Channel ID

FDM Recon SQL.Gl Account Id

FDM Recon SQL.Org Unit Id

FDM Recon SQL.Transaction ID

Calculations

Item Name	Formula	Description
IDENTITY_CODE	User-defined value	Each month, the administrator should change the Identity Code to a unique number for tracking purposes.
INTEREST_RATE_CD	User-defined value	Choose a number that helps identify the transaction, such as a user-defined number.
ORIGINATION_DATE	User-defined value	The default strategy is to subtract one month from the month being reconciled.
AS_OF_DATE	User-defined value	A Not Null columns in the OFSA_Ledger_Stat_Recon table that must be populated. Set the date for the last day of the month being reconciled.
CUR_BOOK_BAL	Lsbal SUM - Ibal SUM	This column is the difference of the Ledger_Stat balance minus the Instrument balance. This is the number that gets written to the OFSA_Ledger_Stat_Recon table.
Ibal SUM	SUM(CUR_BOOK_BAL) of Instrument tables	Instrument balance
Lsbal SUM	SUM(MONTH_12) of Ledger Stat	Ledger Stat balance

Filter
CUR_BOOK_BAL <> 0 (eliminate zeroes)

Join
GL Account ID Desc > Recon SQL

11.4.2 Ledger Stat Reconciliation Upload Report

The Ledger Stat Reconciliation Upload Report shows all out-of-balance conditions between the instrument and Ledger_Stat tables. Also, the workbook provides a method to post the out of balance conditions. Put another way, this report shows all outages between the Ledger Stat table and the Instrument table and provides a mechanism to plug data if required.

The Reconciliation Report utilizes tables OFSA_Detail_GL_Account_ID and OFSA_ Ledger_Sat_Recon, control file lsrecon.ctl, and CSV file recon.csv file to run. The tables, formerly created by the user, are now permanently seeded during the database creation/upgrade process.

To create and populate the OFSA_DETAIL_GL_ACCOUNT_ID table, refer to the procedures for the Ledger Stat Instrument report in the preceding section.

Folder

FDM Recon SQL

Items

FDM Recon SQL.Common Coa Id

FDM Recon SQL.Org Unit Id

FDM Recon SQL.Gl Account Id

FDM Recon SQL.Channel Id

FDM Recon SQL.Transaction Id

FDM Recon SQL.Cust Seg Id

Calculations

Item Name	Formula	Description
AS_OF_DATE	User-defined value	A NOT NULL column in the OFSA_LEDGER_ STAT_RECON table that must be populated. Set the date to the last day of the month being reconciled.
CUR_BOOK_BAL	Lsbal SUM - Ibal SUM	This column is the difference of the Ledger_ Stat balance minus the Instrument balance. This is the number that gets written to the OFSA_LEDGER_STAT_RECON table.

Filter

CUR_BOOK_BAL <> 0

11.4.2.1 Indexing the OFSA_Ledger_Stat_Recon Table

Note that the database administrator needs to manually maintain the indexing of the OFSA_LEDGER_STAT_RECON table for maximum performance.

Be sure to note which columns are not null. These columns MUST be included in the SQL statement (discussed later) as select and group by items.

11.4.2.2 Preparing Discoverer Folders

This section explains the needed modifications for the SQL statement used to generate the Recon Report.

1. Using Discoverer Administration Edition, right click on the folder labeled Recon SQL in the FDM business area and choose Properties from the right click menu.

2. Click the band labeled Custom SQL to call up the complete SQL statement. Here the administrator makes the required changes to run this report.

3. Be sure that each instrument table is represented in this SQL statement.

4. If it is not, copy an existing statement (starting with select and ending with UNION), paste it above the Ledger_Stat area and modify the "From" clause by replacing it with the new instrument table name.

5. Include the name of the new table in the where clause at the end of the SQL statement.

6. If there is a table in the statement that does not exist in the database, be sure to delete that section. Also, remove the table's name from the where clause at the end of the SQL statement.

7. Include any user-defined leaves in the select and group by areas.

8. Be sure there is a section for the OFSA_Ledger_Stat_Recon table. The statement is exactly the same as the other instrument tables, except the "From" needs to be OFSA_Ledger_Stat_Recon.

9. Every month, the administrator needs to change the AS_OF_DATE and the Ledger_Stat month bucket. Also, at the beginning of each year, YEAR_S needs to be changed.

11.4.2.3 SQL Excerpt

The following is a excerpt from the SQL statement contained in the Recon SQL folder.

```
SELECT COMMON_COA_ID, ORG_UNIT_ID,GL_ACCOUNT_ID, SUM(CUR_BOOK_BAL) IBAL, 0 AS
LSBAL
FROM MORTGAGES
WHERE AS_OF_DATE = '31-DEC-1997'
GROUP BY COMMON_COA_ID, ORG_UNIT_ID, GL_ACCOUNT_ID
UNION
...
UNION
(SELECT COMMON_COA_ID, ORG_UNIT_ID, GL_ACCOUNT_ID, 0,
    SUM(MONTH_12) LSBAL
FROM LEDGER_STAT
WHERE FINANCIAL_ELEM_ID = 100
    AND CONSOLIDATION_CD = 100
    AND YEAR_S = '1999'
    AND ACCUM_TYPE_CD='D'
    AND GL_ACCOUNT_ID IN
    (SELECT DISTINCT GL_ACCOUNT_ID
    FROM OFSA_DETAIL_GL_ACCOUNT_ID
    WHERE TABLE_NAME IN ('MORTGAGES', 'COMMERCIAL_LOAN', 'CONSUMER_LOAN',
    'CREDIT_CARDS', 'MORTGAGE_BACK_SEC', 'WHOLESALE_FUNDING', 'INVESTMENTS' ,
    'DEPOSITS')
GROUP BY COMMON_COA_ID, ORG_UNIT_ID, GL_ACCOUNT_ID)
```

11.4.2.4 Running the Workbook

The procedures for preparing and running the Ledger Stat Reconciliation Workbook are:

1. Open the Ledger Stat Reconciliation Report workbook, recon.dis.

2. Set the above conditions and calculations in the workbook.

3. Run the Upload worksheet and export the outages to a CSV file.

11.4.2.5 Worksheets

The Ledger Stat Reconciliation Report workbook contains two worksheets: The Audit Report and the Upload Report.

11.4.2.5.1 Audit Report The Audit Report is used for auditing purposes. The user sees which accounts are out of balance between the instrument and Ledger_Stat tables.

Run the Audit worksheet in the report. If you plan to write back the changes, continue with the following directions. If this report is used only for showing the outages, stop here. Note: If desired, a hierarchy table works with this report to give drilling capability.

11.4.2.5.2 Upload Report The Upload report provides data to write back the differences to the OFSA_Ledger_Stat_Recon table.

Run the Upload worksheet in the report. Save the Upload report as an .CSV file. To do so, choose export and then the drop down box to choose "CSV(Comma Delimited)(*.csv)." This becomes the data file that writes back to the OFSA_Ledger_Stat_Recon table.

11.4.2.6 Populating the OFSA_Ledger_Stat_Recon table from the recon.csv file

This section explains how to populate the OFSA_Ledger_Stat_Recon table. Follow these steps only if you are planning to reconcile the data by plugging the OFSA_Ledger_Stat_Recon table.

Make the following modifications in the lsrecon.ctl file, which is located in the OFSA_INSTALL/dbs/<OFSA release> /utilities/reporting directory on the server side, or in the <Oracle Home>\<OFSA release>\disco31\ directory on the client side.

1. Modify the INFILE name to include the location of the CSV file from above.

2. Include any customer-defined leaves in the list of columns at the bottom of the file. Again, remember that all not null columns in the OFSA_Ledger_Stat_Recon table must be included.

> **Note:** Id_number sequence (1,1) (This automatically populates this column with a unique number for each row. Id_number must be unique.)

3. Call the SQL Loader and run the process. This loads the data from the .csv file to the OFSA_Ledger_Stat_Recon table.

11.4.2.7 Rerunning the Workbook

After finishing the above, rerun the Audit worksheet in the workbook. This time, since data is in the OFSA_Ledger_Stat_Recon table, the Ledger_Stat and Instrument tables should balance.

12

Performance Analyzer Reports for the Banking and Lending Industry

This chapter describes the reporting capabilities of Oracle Performance Analyzer as they apply to the banking and lending industry.

Oracle Performance Analyzer reports deliver a comprehensive profitability picture across a financial organization based on allocation activities. These reports support multicurrency reporting, enabling multinational corporations to analyze financial performance in different currencies. These reports include:

Allocation Description Report

Audit Trail Report

Balance Sheet Report

Balance Sheet Multicurrency Report

Balance Sheet and Income Statement Report

Balance Sheet and Income Statement Multicurrency Report

Income Statement Report

Income Statement Multicurrency Report

Instrument/Transaction Drill-to-Detail Reports

Ledger Stat Drill-to-Detail Reports

Single Stratification Report

Single Stratification Multicurrency Report

Single Stratification User Description Report

Single Stratification User Description Multicurrency Report

Double Stratification Report

Double Stratification Multicurrency Report

Double Stratification User Description Report

Double Stratification User Description Multicurrency Report

12.1 Allocation Description Report

The Allocation Description Report displays the Allocation ID documentation.

Workbook
allocate.dis

Sheets
Allocation Description

Batch List

Data Filter ID List

Tree Filter ID List

Table ID List

Lookup Table ID List

12.1.1 Allocation Description Sheet

The Allocation Description Sheet contains the following components:

Folders
PA OFSA Alloc Leaves

PA OFSA Catalog Of IDs

PA OFSA IDT Alloc

PA Row Num Order

Items
PA OFSA Alloc Leaves.Leaf Num Id

PA OFSA Alloc Leaves.Leaf Node

PA OFSA Alloc Leaves.Tree Sys Id

PA OFSA Alloc Leaves.Constant Char

PA OFSA Alloc Leaves.Data Filter Sys Id

PA OFSA Catalog Of Ids.Id Desc Short

PA OFSA Catalog Of Ids.Folder Name

PA OFSA Idt Alloc.Sys Id Num

PA Row Num Order.Row Number

Calculations

Table 12–1   Allocation Description Sheet Calculations

Item Name	Formula	Description
Allocation Page	PA OFSA IDT Alloc.Sequences+1	In the database, the Allocation page begins at sequence 0. Increment Sequences by 1 to be consistent with Allocation interface screen.
Audit Trail	DECODE(Audit Trail Flag, 0, 'NO', 1, 'YES')	Return appropriate text based on Audit Trail Flag values of the OFSA_Audit_Trail table.
Leaf Description	DECODE(PA OFSA Alloc Leaves.Leaf Num ID, -1, DECODE(Alloc Dim Subtype Cd,2,'ISO Curr CD',Factor'),0,'Fin Elem ID',1,Org Unit ID',2,GL Acct ID',3,'Common COA', 7,'Function', 8,Element',9, 'Actual GL Account ID',10,'Transaction Type')	Return appropriate text based on Leaf Num ID values of the OFSA_Alloc Leaves table.
Row Type Desc	DECODE(Row Type, 10, 'Debit', 20, 'Credit', 30, 'Filter On', 40, 'Percent of'', 45, 'Percent of No Total', 50, 'Leaf', 100, 'Plus', 110, 'Minus', 120, 'Multiply', 130, 'Divide', 200, 'Constant', 300, 'Table ID', 350, 'Lookup Table ID',400, 'Macro', 410, 'Field', 500, 'Right Parenthesis', 550, 'Left Parenthesis', Row Type)	Return appropriate text based on Row Type values of the OFSA_ROW_NUM_ORDER table.
Leaf Node Desc	DECODE(Leaf Node,0,'',-99100,'None',-99300,'Same as Filter',-99200, DECODE(PA OFSA Alloc Leaves.Leaf.Num ID,-1, DECODE(Alloc Dim Subtype Cd,1,'',''All'),'All'), -99400,'Same as%Distr',-99410,'Same as%Distr-Lf',-99450,'Same as Table',-99600, DECODE(Alloc Dim Subtype, Cd,2,Dimension Value Character,Leaf Node),Leaf Node)	Shows text description of leaf macros or the specified leaf number. The node number is displayed if a Tree Filter ID is used.
Data Filter Sys ID	DECODE(Data Filter Sys ID,-99100,'None',0,'',Data Filter Sys ID)	Shows the Sys ID of the Data Filter, if present.

Table 12–1   (Cont.) Allocation Description Sheet Calculations

Item Name	Formula	Description
Tree Sys ID	DECODE(Tree Sys ID,0,'',-99100,'',-99500,'Tree Filter',Tree Sys ID)	Shows the Sys ID of a rollup or Tree Filter if it is a Tree Filter.
Server Process	DECODE(Process Target,0,'NO',1,'YES')	Return appropriate text based on Process Target values of the OFSA_IDT_ALLOC table.
Single Pass	DECODE(Parallel Flag, 0, 'No',1,'YES')	Return appropriate text based on Parallel Flag values of the OFSA_IDT_ALLOC table.

Joins

OFSA IDT Alloc > OFSA Alloc Leaves

OFSA Catalog Of IDs > OFSA IDT Alloc

OFSA Alloc Leaves > Row Num Order

12.1.2 Batch ID List Sheet

The Batch ID List Sheet has the following components:

Folders

PA OFSA Catalog of IDs

PA OFSA IDT Batch

PA OFSA catalog of IDs Alloc Names

Items

PA OFSA Catalog of IDs.ID Des Short

PA OFSA Catalog of IDs.Folder Name

PA OFSA IDT Batch Allocations Sys ID Num

PA OFSA IDT Batch Sequences

PA OFSA IDT Alloc.Sys ID. Num

PA OFSA Catalog of IDs Alloc Names ID Name

Calculations

Item Name	Formula	Description
Batch Application	DECODE(PA OFSA Catalog of IDs.Chg App,0, 'Performance Analyzer',2'Risk Manager',3,'Transfer Pricing',4,'Balance and Control',5,'Portfolio Analyzer', Customer Houseolding','7,'Market Manager',8, Discoverer Integrator',9,'Rate Manager','Other')	Shows the proper type of Batch ID.

Joins

OFSA Catalog of IDs > OFSA IDT Batch

OFSA IDT Batch > OFSA Catalog of IDs Alloc Names

12.1.3 Data Filter ID List Sheet

The Data Filter ID List Sheet has the following components:

Folder

PA OFSA Catalog of IDs.

Items

PA OFSA Catalog of IDs.ID Desc Short

PA OFSA Catalog of IDs.Folder Name

PA OFSA Catalog of IDs.Sys ID. Num

Condition

ID Type=4

12.1.4 Tree Filter ID List Sheet

The Tree Filter ID List Sheet has the following components:

Folder
PA OFSA Catalog of IDs

Items
PA OFSA Catalog of IDs.ID Desc Short

PA OFSA Catalog of IDs.Folder Name

PA OFSA Catalog of IDs.Sys ID. Num

Condition
ID Type=8

12.1.5 Table ID List Sheet

The Table ID List Sheet has these components:

Folder
PA OFSA Catalog of IDs

Items
PA OFSA Catalog of IDs.ID Desc Short

PA OFSA Catalog of IDs.Folder Name

PA OFSA Catalog of IDs.Sys ID. Num

Condition
ID Type=17

12.1.6 Lookup Table ID List Sheet

The Lookup Table ID List Sheet has these components:

Folders

PA OFSA Catalog of IDs

Items

PA OFSA Catalog of IDs.ID Desc Short

PA OFSA Catalog of IDs.Folder Name

PA OFSA Catalog of IDs.Sys ID. Num

Condition

ID Type=31

12.2 Audit Trail Report

The Audit Trail Report runs against the OFSA_Audit_Trail table, which contains all of the audit information for all allocations that have been performed.

Workbook

audit.dis

Folders

PA Audit Trail

PA Common COA ID Desc

PA Financial Elem ID Desc

PA GL Account ID Desc

PA Organizational Unit ID Desc

Items

PA Audit Trail.Amount

PA Audit Trail.As Of Date

PA Audit Trail.Identity Code

PA Audit Trail.Link Key

PA Audit Trail.Volume Rate

PA Common COA ID Desc.Common COA ID Desc

PA Financial Elem ID Desc.Fin Elem ID Desc

PA GL Account ID Desc.GL Acct ID Desc

PA Organizational Unit ID Desc.Org ID Desc

Calculations

Item Name	Formula	Description
Allocation ID	Alloc Sys Id||' '||Id Desc Short	Returns concatenated string joining Allocation Sys ID and ID Short Description.
To/From	DECODE(To From Flag,'T','To','F','From')	Returns appropriate text based on To From Flag values.

Sort On

PA Audit Trail.Link Key

Calculated Item - To/From

Joins

GL Account ID Desc > Audit Trail

Organizational Unit ID Desc > Audit Trail

Catalog Of Ids > Audit Trail

Common COA ID Desc > Audit Trail

Financial Elem ID Desc > Audit Trail

12.3 Balance Sheet Report

The Balance Sheet Report shows exactly where a bank's asset, liability, and equity distribution stands at any given time.

Workbook

balance.dis

Folders

Std Rpt Hier-GL Account Hier

Std Rpt Hier-Org Unit Hier

Std Rpt Ledger Stat Reporting

Items

Std Rpt Hier-GL Account Hier.Level 1 Description
(returns Equity, Liability, and Asset level information)

Std Rpt Hier-GL Account Hier.Level 2 Description
(returns the next level below Equity, Liability, and Asset)

Std Rpt Hier-Org Unit Hier.State Banks Description

Std Rpt Hier-Org Unit Hier.Total Rollup Description

Std Rpt Ledger Stat Reporting.End Date

Std Rpt Ledger Stat Reporting.End Balance

Calculations

Item Name	Formula	Description
End Balance	SUM(DECODE("Level 1 Description", 'Assets',End Balance/1000,'Liabilities and Equity',End Balance/1000,'0'))	Finds all GL Account Leaf Values that fall under either the Assets or Liabilities and Equity labels on level 1 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000 for reporting purposes.

Filters

"Channel Value <> '-99000L01' (Eliminates any orphan leaves.)

"Level 6 Value <> '-99000L01' (Eliminates any orphan leaves.)

Consolidation Code = 100

End Date = :End Date

Level 1 Description IN('Liabilities and Equity','Assets')

This last filter reports only liabilities, equity, and assets. You should substitute the appropriate descriptions based on their rollup.

Joins

OFS_RPT_HIER_GL_ACCT_ID: ORACLE (GL_ACCOUNT_ID)
> Std Rpt Ledger Stat Reporting

OFS_RPT_HIER_ORG_UNIT: ORACLE (ORG_UNIT_ID)
> Std Rpt Ledger Stat Reporting

User Prompt

The user prompt is End Date. Enter the end dates to report on.

12.4 Balance Sheet Multicurrency Report

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency specified by the user. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features."

Workbook

balance_mc.dis

12.5 Balance Sheet and Income Statement Report

This report displays the institutional high-level finances on one page.

Workbook

inc-bal.dis

Folders

Std Rpt Hier-GL Account Hier

Std Rpt Hier-Org Unit Hier

Std Rpt Ledger Stat Reporting

Items

Std Rpt Ledger Stat Reporting.End Balance

Std Rpt Ledger Stat Reporting.End Date

Std Rpt Ledger Stat Reporting.Interest

Axis Items

Table 12–2   Balance Sheet and Income Statement Report Axis Items

Folder	Item	Description
Std Rpt Hier-GL Account Hier	Level 1 Description	With this hierarchy, Level 1 will return Equity, Liability, and Asset level information. When mapping, be sure to choose the level in the hierarchy that contains Equity, Liability, and Asset or similar items.
Std Rpt Hier-GL Account Hier	Level 2 Description	With this hierarchy, Level 2 will return Earning & Non-earning assets. When mapping, be sure to choose the appropriate level in the hierarchy.
Std Rpt Hier-GL Account Hier	Level 3 Description	With this hierarchy, Level 3 will return Net Margin and Net Non-Interest Income. When mapping, be sure to choose the level in the hierarchy that contains these or similar items.

Table 12–2   (Cont.) Balance Sheet and Income Statement Report Axis Items

Folder	Item	Description
Std Rpt Hier-GL Account Hier	Level 4 Description	With this hierarchy, Level 4 will return Interest Expense, Interest Income, Total Non-Interest Income, and Total Non-Interest Expense. When mapping, be sure to choose the level in the hierarchy that contains these or similar items.
Std Rpt Hier-Org Unit Hier	State Banks Description	State Bank Desc is the next org level down from the Total Rollup Description.
Std Rpt Hier-Org Unit Hier	Total Rollup Description	Total Rollup shows the highest level for the rollup.

Calculations

Table 12–3   Balance Sheet and Income Statement Report Calculations

Item Name	Formula	Description
ASSETS	SUM(NULL)	Label.
BALANCE SHEET ($000,000S)	SUM(NULL)	Label.
INCOME STATEMENT ($000s)	SUM(NULL)	Label.
INCOME	SUM(NULL)	Label.
Interest Expense	SUM(DECODE("Level 4 Description",'Interest Expense',-Interest/1000,0))	Finds all GL Account Leaf Values that fall under the Interest Expense label on level 4 of the chosen hierarchy. Then it subtracts all interest balances in Ledger Stat for these selected GL Account Leaf Values and divides the interest amount by 1,000 for reporting purposes.
Interest Expense Rate	DECODE(Sum(End Balance),0,0,SUM(Decode ("Level 4 Description", 'Interest Expense', -Interest,0)) / SUM(Decode("Level 3 Description",'Total Demand Deposits','End Balance','Total Interest Bearing Funds','End Balance,0))*12)	Divides Interest Expense calculation by Total Deposits calculation and multiplies by 12. Change the 12 if the institution uses a different amortization method.

Table 12–3   (Cont.) Balance Sheet and Income Statement Report Calculations

Item Name	Formula	Description
Interest Income	SUM(DECODE("Level 4 Description",'Interest Income',Interest/1000,0))	Finds all GL Account Leaf Values that fall under the Interest Income label on level 4 of the chosen hierarchy. Then it sums all interest balances in Ledger Stat for these selected GL Account Leaf Values and divides the interest amount by 1,000 for reporting purposes.
LIABILITIES:	SUM(NULL)	Label.
Net Income	SUM(DECODE("Level 4 Description",'Total Non-Interest Income',Non Interest Income/1000,0)) -SUM(DECODE("Level 4 Description",'Total Non-interest Expense',Non Interest Expense/1000,0)) +SUM(DECODE(" Level 4 Description",'Interest Income',Interest/1000,0))-SUM(DECODE("Level 4 Description",'Interest Expense',Interest/1000,0))	Subtracts Total Non-Interest Expense calculation from Total Non-Interest Income calculation and adds it to Interest Income calculation minus Interest Expense calculation.
Net Margin	SUM(DECODE("Level 4 Description",'Interest Income',Interest/1000,0))-SUM(DECODE("Level 4 Description",'Interest Expense',Interest/1000,0))	Subtracts Interest Expense calculation from Interest Income calculation.
Net Non-Interest Income	SUM(DECODE("Level 4 Description",'Total Non-Interest Income',Non Interest Income/1000,0))-SUM(DECODE ("Level 4 Description",'Total Non-interest Expense',Non Interest Expense/1000,0))	Subtract the Total Non-Interest Expense calculation from the Total Non-Interest Income calculation.
Non-Earning Assets	SUM(DECODE("Level 2 Description",'Non-Earning Assets',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Non-Earning Assets label on level 2 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Security Gain/Loss	SUM(Decode("Level 3 Description",'Security Gain/Loss', End Balance/1000000,0))	Returns the appropriate values with correct signage for gains and losses.

Table 12–3   (Cont.) Balance Sheet and Income Statement Report Calculations

Item Name	Formula	Description
Total Assets	SUM(DECODE("Level 1 Description",'Assets',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Assets label on level 1 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Total Demand Deposits	SUM(DECODE("Level 3 Description",'Total Demand Deposits',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Total Demand Deposits label on level 3 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Total Earning Assets	SUM(DECODE("Level 2 Description",'Earning Assets',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Earning Assets label on level 2 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Total Interest Bearing Funds	SUM(DECODE("Level 3 Description",'Total Interest Bearing Funds',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Total Interest Bearing Funds label on level 3 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Total Liabilities	SUM(DECODE("Level 2 Description",'Total Liabilities',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under the Total Liabilities label on level 2 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.

Table 12–3   (Cont.) Balance Sheet and Income Statement Report Calculations

Item Name	Formula	Description
Total Non-Interest Expense	SUM(DECODE("Level 4 Description",'Total Non-interest Expense',-Non Interest Expense/1000,0))	Finds all GL Account Leaf Values that fall under the Total Non-Interest Expense label on level 4 of the chosen hierarchy. Then it subtracts all Non-Interest balances in Ledger Stat for these selected GL Account Leaf Values and divides the interest amount by 1,000 for reporting purposes.
Total Non-Interest Income	SUM(DECODE("Level 4 Description",'Total Non-Interest Income',Non Interest Income/1000,0))	Finds all GL Account Leaf Values that fall under the Total Non-Interest Income label on level 4 of the chosen hierarchy. Then it sums all Non Interest Income balances in Ledger Stat for these selected GL Account Leaf Values and divides the Non Interest Income balance amount by 1,000,000 for reporting purposes.
Total Other Liabilities	SUM(DECODE("Level 3 Description",'Acceptances Outstanding',End Balance/1000000,'Other Liabilities',End Balance/1000000,0))	Finds all GL Account Leaf Values that fall under either the Acceptances Outstanding or Other Liabilities label on level 3 of the chosen hierarchy. Then it sums all end balances in Ledger Stat for these selected GL Account Leaf Values and divides the end balance amount by 1,000,000 for reporting purposes.
Yield on Earning Assets	DECODE(SUM(End Balance),0,0,SUM(DECODE("Level 4 Description",'Interest Expense',Interest))/SUM(DECODE("Level 2 Description","Earning Assets',End Balance))*12)	Divides Interest Expense calculation by Earnings Assets calculation and multiplies the result by 12. Change the 12 if the institution uses a different amortization method.

Filters

Filter Name	Formula	Description
Orphan Filter	Level 6 Value <> '-99000L01'	This orphan filter is generated by Discoverer Integrator to eliminate any orphan numbers. It can be turned on or off.
Consolidation Code	Consolidation Code = 100	This will return only actual numbers, not budgeted nor forecasted. It can be turned on or off.
Level 1 description	UPPER('Level 1 Description') IN (UPPER('Net Income'),UPPER('Assets'),UPPER(' Liabilities and Equity'))	This eliminates anything other than Liabilities, Equity and Assets. The use of UPPER coupled with match case in the check box in the condition wizard speeds up the query.

Joins

OFS_RPT_HIER_GL_ACCT_ID: ORACLE (GL_ACCOUNT_ID)
> Std Rpt Ledger Stat Reporting

User Prompt

The prompt is End Date. Enter one or more end dates to report on.

12.6 Balance Sheet and Income Statement Multicurrency Report

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency specified by the user. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features," for usage of the balance conversion function.

Workbook

inc-bal_mc.dis

12.7 Income Statement Report

The Income Statement Report displays profitability at any given moment.

Workbook
income.dis

Folders
Std Rpt Hier-GL Account Hier

Std Rpt Hier-Org Unit Hier

Std Rpt Ledger Stat Reporting

Items
Std Rpt Hier-GL Account Hier.Level 3 Description

Std Rpt Hier-GL Account Hier.Level 4 Description

Std Rpt Hier-Org Unit Hier.State Banks Description

Std Rpt Hier-Org Unit Hier.Total Rollup Description

Std Rpt Ledger Stat Reporting.End Date

Std Rpt Ledger Stat Reporting.Interest

Std Rpt Ledger Stat Reporting.Non Interest Expense

Std Rpt Ledger Stat Reporting.Non Interest Income

Axis Items

Folder	Item	Description
Std Rpt Hier-GL Account Hier	Level 4 Description	With this hierarchy, Level 4 will return Interest Expense, Interest Income, Loan Loss Provision, Total Non-Interest Income, Total Non-Interest Expense and Security Gain/Loss. When mapping, be sure to choose the level in the hierarchy that contains these items.
Std Rpt Hier-GL Hier	Level 3 Description	With this hierarchy, Level 3 will return Net Margin, Net Non-Interest Income and Security Gain/Loss. When mapping, be sure to choose the level in the hierarchy that contains these types of items.
Std Rpt Hier-Org Unit Hier	State Banks Description	State Bank Desc is the next org level down from the Total Rollup Description.
Std Rpt Hier-Org Unit Hier	Total Rollup Description	Total Rollup shows the highest level for the chosen org rollup.

Calculations

Item	Formula	Description
Income	SUM(Signage * Interest - Non Interest Expense + Non Interest Income)/1000	Multiplies the signage calculation by the interest and expense items to allow for proper signage. The formula then divides the figures by 1000 for correct reporting.
Signage	DECODE('Level 4 Description', 'Interest Expense', -1, 'Total Non-Interest Expense', -1, 'Loan Loss Provision',-1,1)	Multiplies the expenses by negative one so the expenses appear with a negative sign.

Filters

Filter Name	Formula	Description
Orphan Filter	Level 6 Value <> '-99000L01'	This orphan filter is generated by Discoverer Integrator to eliminate any orphan numbers. It can be turned on or off.
Orphan Filter	Channel Value <> '-99000l01'	This orphan filter is generated by Discoverer Integrator to eliminate any orphan numbers. It can be turned on or off.
N/A	"Level 3 Description" IN ('Net Margin','Net Non-Interest Income','Security Gain/Loss')	Level 3 will return all values for Net Margin, Net Non-Interest Income and Security Gain/Loss. The user should substitute the appropriate descriptions based on the organization's rollup.
N/A	Consolidation Code = 100	This will return only actual numbers, not budgeted nor forecasted.

Joins

OFS_RPT_HIER_GL_ACCT_ID: ORACLE (GL_ACCOUNT_ID)
> Std Rpt Ledger Stat Reporting

OFS_RPT_HIER_ORG_UNIT: ORACLE (ORG_UNIT_ID)
> Std Rpt Ledger Stat Reporting

User Prompt

The prompt is End Date. Enter the end dates to report on.

12.8 Income Statement Multicurrency Report

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency specified by the user. (Refer to "Registering PL/SQL Functions for Multicurrency"in Chapter 4, "Implementing Advanced FDM Reporting Features" for usage of the balance conversion function.)

Workbook

income_mc.dis

12.9 Instrument/Transaction Drill-to-Detail Reports

The Instrument/Transaction Drill-to-Detail Reports are prototypes that demonstrate the drill capabilities between an instrument table and a transaction table.

The instrument and transaction tables are joined by Identity Code and ID Number. By creating a drill-to-detail between the Instrument table and the Transaction table based on ID Number, you can double-click the ID number and return all the data that makes up the balance for the ID number. Depending on your database, there may be multiple identity codes returned as well. Be sure to apply a condition to limit the report to the specific identity code in the original report. This drill enables you to see all the transaction types that make up the balance in the instrument table.

Workbook

instrans.dis

Reports

Instrument Table Report

Drill Information Report

12.9.1 Instrument Table Report

The Instrument Table Report contains instrument information and demonstrates how to set up drills between an instrument table and a transaction table. This report has the following components:

Folder

Deposit

Items

Deposit.As Of Date

Deposit.Common Chart Of Accounts Id

Deposit.Current Net Book Balance SUM

Deposit.General Ledger Account Id

Deposit.Id Number

Deposit.Identity Code

Deposit.Organizational Unit Id

Filter

ID Number IN ('5949','17816','38076')

12.9.2 Drill Information Report

The Drill Information Report contains the transaction information and demonstrates how to set up drills between an instrument table and a transaction table. This report contains the following components:

Folder

Trans Volume Loan

Items

Trans Volume Loan.As Of Date

Trans Volume Loan.As Of Date Year

Trans Volume Loan.As Of Date Quarter

Trans Volume Loan.As Of Date Month

Trans Volume Loan.As Of Date Day

Trans Volume Loan.Common Coa Id

Trans Volume Loan.Gl Account Id

Trans Volume Loan.Id Number

Trans Volume Loan.Identity Code

Trans Volume Loan.Identity Code Chg

Trans Volume Loan.Iso Currency Cd

Trans Volume Loan.Org Unit Id

Trans Volume Loan.Other 01 Cost

Trans Volume Loan.Other 02 Cost

Trans Volume Loan.Other 03 Cost

Trans Volume Loan.Other 04 Cost

Trans Volume Loan.Other 05 Cost

Trans Volume Loan.Other 06 Cost

Trans Volume Loan.Other 07 Cost

Trans Volume Loan.Other 08 Cost

Trans Volume Loan.Other 09 Cost

Trans Volume Loan.Other 10 Cost

Trans Volume Loan.Other 11 Cost

Trans Volume Loan.Other 12 Cost

Trans Volume Loan.Record Count

Trans Volume Loan.Total Cost

Trans Volume Loan.Trans Id

Trans Volume Loan.Trans Vol

Trans Volume Loan.True Gl Id

Filter
ID Number = 5949

12.10 Ledger Stat Drill-to-Detail Reports

The Ledger Stat Drill-to-Detail reports provide different dimensions on Ledger Stat and Instrument tables.

Workbook
lidrill.dis

Reports
Instrument Drill Report

Ledger Stat Drill Report

12.10.1 Instrument Drill Report

The Instrument Drill Report contains instrument information. It demonstrates how to set up drills between the Ledger Stat and an Instrument table, and vice versa. This report has the following components:

Folder

Std Rpt Instrument

Items

Std Rpt Instrument.General Ledger Account Id

Std Rpt Instrument.Organizational Unit Id

Std Rpt Instrument.Common Chart Of Accounts Id

Std Rpt Instrument.As Of Date

Std Rpt Instrument.Current Net Book Balance SUM

Filter

Organizational Unit Id IN ('5013964','5014064','5014164','5710061')

12.10.2 Ledger Stat Drill Report

The Ledger Stat Drill Report contains Ledger Stat information and demonstrates how to set up drills between the Ledger Stat and Instrument table.

Folder

Std Rpt Ledger Stat Reporting

Items

Std Rpt Ledger Stat Reporting.End Balance SUM

Std Rpt Ledger Stat Reporting.End Date

Std Rpt Ledger Stat Reporting.Organizational Unit Id

Std Rpt Ledger Stat Reporting.General Ledger Account ID

Std Rpt Ledger Stat Reporting.Common Chart Of Accounts ID

Sort On

Std Rpt Ledger Stat Reporting.Organizational Unit ID

Filters

General Ledger Account Id IN ('37591','37601')

Organizational Unit Id IN ('5013964','5014064','5014164','5710061')1

12.11 Single Stratification Report

This report contains one stratification tier. It also allows the user to specify both the X and Y axis items, which can be moved at the discretion of the user.

Workbook
strat1.dis

Folder
Std Rpt Instrument

Items
Std Rpt Instrument.Current Gross Rate

Std Rpt Instrument.Current Net Book

Std Rpt Instrument.Id Number

Std Rpt Instrument.Identify Code

Std Rpt Instrument.Maturity Date

Calculations

Item	Formula	Description
Tier 1 Calculation	DECODE(LEAST(GREATEST(Current Gross Rate,:T1L),:T1H),Current Gross Rate,'01. '||:T1L||'-'||:T1H	This decode statement looks for the lowest rate entered by the user (designated by :T1L, a parameter,) and for the greatest number (designated by :T1H, another parameter). The numbers reference the Current Gross Rate item and then print to screen the entered numbers. (The Desc report prints whatever the user enters as the description, not the values.) By replacing the Current Gross Rate with another item, you can query on an item in the database. For example, if you replace it with Current Book Balance, you can specify a range of balances for a report.

12.12 Single Stratification Multicurrency Report

The Single Stratification Multicurrency Report performs the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency that you selected.

Refer to "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features" for usage of the balance conversion function.

Workbook

strat1_mc.dis

12.13 Single Stratification User Description Report

The Single Stratification User Description Report functions the same way as the Single Stratification Report. You can enter a description for each tier item, which makes this report easier to use.

Workbook

strat1d.dis

12.14 Single Stratification User Description Multicurrency Report

Th Single Stratification User Description Multicurrency Report functions the same as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency that you selected.

Refer to "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features" for usage of the balance conversion function.

Workbook

strat1d_mc.dis

12.15 Double Stratification Report

This report has two tiers. You can specify both the X and Y axis items, and move them at will.

Workbook

strat2.dis

Folder

Std Rpt Instrument

Items

Std Rpt Instrument.Current Gross Rate

Std Rpt Instrument.Current Gross Book Balance

Std Rpt Instrument.Id Number

Std Rpt Instrument.Identify Code

Std Rpt Instrument.Maturity Date

Calculations

Table 12–4   Double Stratification Report Calculations

Item	Formula	Description
Tier 1 Calculation	DECODE(LEAST(GREATEST(Current Gross Rate,:T1L),:T1H),Current Gross Rate,'01. '||:T1L||'-'||:T1H	This decode statement looks for the lowest rate entered by the user (designated by :T1L, a parameter,) and for the greatest number (designated by :T1H, another parameter). The numbers reference the Current Gross Rate item and then prints to screen the entered numbers. (The Desc report prints whatever the user enters as the description, not the values.) By replacing the Current Gross Rate with another item, the user can query on an item in the database. For example, replace it with Current Book Balance, and the user can specify a range of balances for a report.

Table 12–4   (Cont.) Double Stratification Report Calculations

Item	Formula	Description
Tier 2 Calculation	DECODE(LEAST(GREATEST(Maturity Date,:T1L),:T1H),Maturity Date,'01.'||:T1L||'-'||:T1H	This decode statement looks for the lowest rate entered by the user (designated by :T1L, a parameter,) and for the greatest number (designated by :T1H, another parameter). The numbers reference the Maturity Date item and then prints to screen the entered numbers. (The Desc report prints whatever the user enters as the description, not the values.) By replacing the Maturity Date with another item, the user can query on an item in the database. For example, replace it with Current Book Balance, and the user can specify a range of balances for a report.

12.16 Double Stratification Multicurrency Report

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency that you specified. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features."

Workbook

strat2_mc.dis

12.17 Double Stratification User Description Report

The Double Stratification User Description Report functions in the same manner as the Double Stratification Report **(strat2.dis)**. Enter a description for each tier item to make the report easier to use.

Workbook

strat2d.dis

12.18 Double Stratification User Description Multicurrency Report

This report provides the same functions as its single-currency counterpart, except that all balance amounts are automatically converted into the ISO currency specified by the user. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features."

Workbook

strat2d_mc.dis

13

Performance Analyzer Reports for the Insurance Industry

This chapter describes the reporting capabilities of Oracle Performance Analyzer as they apply to the insurance industry. These reports include:

- Asset Share Calculation
- Average Premium
- Lapse Data
- Life Insurance Profitability
- Loss Development
- Loss Trend
- Policies in Force
- Property and Casualty Insurance Profitability

13.1 Asset Share Calculation Report

The Asset Share Calculation report shows an asset share calculation for a block of life insurance policies.

Workbook

pa_ins_assetshare.dis

Preparing to Run the Report

Before running this report, do the following:

1. Create three Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively.

Folders

PA INS Whole UL Life Policies

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS Whole UL Life Policies.Count In Force Lives

PA INS Whole UL Life Policies.Death Benefits Paid

PA INS Whole UL Life Policies.Dividend Amt Month

PA INS Whole UL Life Policies.Fund Bal Per Policy End

PA INS Whole UL Life Policies.Int Earn Cur Month

PA INS Whole UL Life Policies.Policy Age

PA INS Whole UL Life Policies.Prem Gross Amt Month

PA INS Whole UL Life Policies.Withdrawal Amt Gross

PA INS Whole UL Life Policies.Acquisition Costs

PA INS Whole UL Life Policies.Commissions

PA INS Whole UL Life Policies.Exp Allowance

PA INS Whole UL Life Policies.Maintenance Exp

PA INS Whole UL Life Policies.Medical and Inspection

PA INS Whole UL Life Policies.Taxes and Fees

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Life insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Total Expenses	Commissions SUM + Exp Allowance SUM + Medical And Inspection SUM + Maintenance Exp SUM + Acquisition Costs SUM + Taxes And Fees SUM	Sum of various underwriting expenses
Asset Share at End of Year	DECODE(Count In Force Lives SUM,0,0,Fund Bal Per Policy End SUM/Count In Force Lives SUM)	Divides the fund balance for a block of policies by the number of lives in force
Average Reserve Per Policy	DECODE(Count In Force Lives SUM,0,0,Per Policy Reserve End SUM/Count In Force Lives SUM)	Divides the ending policy reserve for a block of policies by the number of lives in force
Surplus at End of Year	Asset Share at End of Year-Average Reserve Per Policy	Computes the amount of policy contribution to surplus

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS Whole UL Life Policies

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS Whole UL Life Policies

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS Whole UL Life Policies

13.2 Average Premium Report

The Average Premium Report shows the number of written exposures and direct
written premiums per month. This report also calculates the average written
premium per policy. Variations of this report include displaying earned or in-force
statistics in lieu of written exposures and premiums. Additionally, a similar report
for each coverage can be produced by redirecting the source to the coverage folder.

Workbook

pa_ins_avgprem.dis

Preparing to Run the Report

Before running this report, do the following:

1. Create two Tree Rollup IDs (one for Common COA and another for Org Unit) in
    Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT and PA_HIER_ORG,
    respectively

Folders

PA INS PL Vehicle Policies

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

Items

PA INS PL Vehicle Policies.As of Date

PA INS PL Vehicle Policies.Exposures Written

PA INS PL Vehicle Policies.Premium Written Direct

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description Page Items

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty or Life insurance products (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written

Calculations

Item	Formula	Description
Average Premium	DECODE(Exposures Written SUM,0,0, Premium Written Direct SUM/Exposures Written SUM)	Divides the total Direct Written Premiums (a dollar amount) by the total number of exposures written (a count number).

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Policies

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Policies

13.3 Lapse Data Report

The Lapse Data life insurance report tracks the number of lives in force, the number of insureds dying and the number of people withdrawing by issue year. This data is typically used to compute retention ratios.

Workbook

pa_ins_lapsedata.dis

Preparing to Run the Report

Prior to using this report, do the following:

1. Create three Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively

Folders

PA INS Whole UL Life Policies

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS Whole UL Life Policies.As of Date

PA INS Whole UL Life Policies.Count Death Incurred

PA INS Whole UL Life Policies.Count Full Surrender

PA INS Whole UL Life Policies.In Force Lives

PA INS Whole UL Life Policies.Issue Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Life insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS Whole UL Life Policies

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS Whole UL Life Policies

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS Whole UL Life Policies

13.4 Life Insurance Profitability Report

The Life Insurance Profitability life insurance report shows loss and expense ratios by issue year and coverage type. The first worksheet summarizes premiums, losses, dividends, and interest earnings. The second worksheet displays the breakdown of various underwriting expenses.

Workbook

pa_ins_lifeprofit.dis

Preparing to Run the Report

Prior to using this report, do the following:

1. Create three Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively

Folders

PA INS Whole UL Life Policies

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS Whole UL Life Policies.Count Death Incurred

PA INS Whole UL Life Policies.Count In Force Lives

PA INS Whole UL Life Policies.Death Benefits Paid

PA INS Whole UL Life Policies.Dividend Amt Month

PA INS Whole UL Life Policies.Int Earn Cur Month

PA INS Whole UL Life Policies.Per Policy Reserve End

PA INS Whole UL Life Policies.Prem Gross Amt Month

PA INS Whole UL Life Policies.Withdrawal Amt Gross

PA INS Whole UL Life Policies.Issue Date Year

PA INS Whole UL Life Policies.Acquisition Costs

PA INS Whole UL Life Policies.Commissions

PA INS Whole UL Life Policies.Exp Allowance

PA INS Whole UL Life Policies.Maintenance Exp

PA INS Whole UL Life Policies.Medical and Inspection

PA INS Whole UL Life Policies.Taxes and Fees

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Life insurance product (for example. each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Total Expenses	Commissions SUM + Exp Allowance SUM + Medical And Inspection SUM + Maintenance Exp SUM + Acquisition Costs SUM + Taxes And Fees SUM	Sum of various underwriting expenses
Loss and Expense Ratio	DECODE(Prem Gross Amt Month SUM,0,0, (Death Benefits Paid SUM+Withdrawal Amt Gross SUM+Per Policy Reserve End SUM+ Dividend Amt Month SUM+Total Expenses-Int Earn Cur Month SUM)∕ Prem Gross Amt Month SUM)	Divides the sum of death benefits paid, withdrawal amounts paid, ending policy reserves, dividend amounts, and total underwriting expenses less interest earnings by the gross premiums collected

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS Whole UL Life Policies

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS Whole UL Life Policies

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS Whole UL Life Policies

13.5 Loss Development Report

The Loss Development report shows accident year development of losses and claim counts. The triangular format of this exhibit is used by actuaries to project future payout and settlement patterns.

Workbook

pa_ins_triangles.dis

Preparing to Run the Report

Prior to using this report, do the following:

1. Create Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively

Folders

PA INS PL Vehicle Claims

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS PL Vehicle Claims.Loss Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty or Life insurance products (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

This workbook contains three similar worksheets. The only difference among the worksheets is that the calculations are shown for paid losses, reported claims, or paid claim counts (that is, the name of the column to be accumulated differs).

Table 13–1   Loss Development Report Calculations

Item	Formula	Description
Paid Loss - period 1	SUM(DECODE(LEAST(SYSDATE,ADD_MONTHS(TO_DATE('12/31/'||TO_CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),:Lat)),SYSDATE,NULL,DECODE(LEAST(As Of Date,ADD_MONTHS(TO_DATE('12/31/'||TO_CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),:Lat)),As Of Date,Loss Amount Paid,0)))	For a selected accident period (typically the 12-month calendar year), this cumulates the total paid losses at a particular point in time. For example, if the parameter, Lat= 6, and the accident year =1998, you will have cumulative paid losses as of June 30, 1999. This column shows cumulative paid losses for various accident years at 18 months old.

Table 13–1   (Cont.) Loss Development Report Calculations

Paid Loss - period 2	SUM(DECODE(LEAST(SYSDATE,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),12+:Lat)),SYSDA TE,NULL,DECODE(LEAST(As Of Date,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),12+:Lat)),As Of Date,Loss Amount Paid,0)))	Continuing with the example above, this column shows cumulative paid losses for various accident years at 30 months old when Lat=6.
Paid Loss - period 3	SUM(DECODE(LEAST(SYSDATE,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),24+:Lat)),SYSDA TE,NULL,DECODE(LEAST(As Of Date,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),24+:Lat)),As Of Date,Loss Amount Paid,0)))	Continuing with the example above, this column shows cumulative paid losses for various accident years at 42 months old when Lat=6.
Paid Loss - period 4	SUM(DECODE(LEAST(SYSDATE,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),36+:Lat)),SYSDA TE,NULL,DECODE(LEAST(As Of Date,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),36+:Lat)),As Of Date,Loss Amount Paid,0)))	Continuing with the example above, this column shows cumulative paid losses for various accident years at 54 months old when Lat=6.
Paid Loss - period 5	SUM(DECODE(LEAST(SYSDATE,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),48+:Lat)),SYSDA TE,NULL,DECODE(LEAST(As Of Date,ADD_ MONTHS(TO_DATE('12/31/'||TO_ CHAR(Loss Date,'YYYY'),'mm/dd/yyyy'),48+:Lat)),As Of Date,Loss Amount Paid,0)))	Continuing with the example above, this column shows cumulative paid losses for various accident years at 60 months old when Lat=6.

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Claims

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Claims

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS PL Vehicle Claims

User Prompt

The user prompt is *Lat*. Enter the number of months since the close of the accident year, in which the loss statistic is to be evaluated.

13.6 Loss Trend Report

The Loss Trend report shows the average cost per claim, the frequency of claims, and pure premiums (loss cost per exposure) over a period of time.

Workbook

pa_ins_losstrend.dis

Preparing to Run the Report

Prior to using this report, do the following:

1. Create three Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively

4. Transform a Ledger Stat table called PA_INS_OFS_RPT_INS that contains the following statistical elements:

   ■ Earned exposures

   ■ Paid loss

   ■ Paid count

   ■ Reported count

Each statistic is assigned a unique Financial Element value. The statistics originate from the monthly load of the instrument table (that is, they are columns in the instrument table) and are assumed to be summarized into the Ledger Stat table every month.

Folders

PA INS OFS RPT INS

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS OFS RPT INS.End Date

PA INS OFS RPT INS.Paid Count

PA INS OFS RPT INS.Reported Count

PA INS OFS RPT INS.Earned Exposures

PA INS OFS RPT INS.Paid Loss

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty or Life insurance products (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, endorsement, or rider associated with the insurance product

Calculations

Item	Formula	Description
Average Cost per Claim	DECODE(Fe500501 SUM,0,0, Fe500504 SUM/Fe500501 SUM)	Fe500501= Paid Count and Fe500504=Losses Paid. Losses paid are divided by the number of claims paid
Incurred Frequency	DECODE(Fe500503 SUM,0,0, (Fe500502 SUM/Fe500503 SUM)*100)	Fe500502= Reported Count and Fe500503=Earned Exposures. The number of claims reported for every 100 earned exposures
Pure Premium	DECODE(Fe500503 SUM,0,0, Fe500504 SUM/Fe500503 SUM)	Fe500503=Earned Exposures and Fe500504=Losses Paid, with the loss cost per earned exposure

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS OFS RPT INS

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS OFS RPT INS

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS OFS RPT INS

13.7 Policies in Force Report

The Policies in Force report shows the number of policies in force (PIF) by distribution channel and inception year. The first worksheet shows the composition of the current book of business. The second worksheet shows the rate at which policies are leaving over time. In practice, an insurance company has many PIF reports based on almost any variable captured in the data model. Similar reports at the coverage level can be produced by redirecting the source to the coverage folder.

Workbook

pa_ins_pif.dis

Preparing to Run the Report

Prior to using this report, do the following:

1. Create two Tree Rollups IDs (one for Common COA and another for Org Unit) in Oracle Performance Analyzer

2. Transform the Tree Rollup IDs using the Transformation ID

3. Name the transformation results, PA_HIER_PRODUCT and PA_HIER_ORG, respectively

Folders

PA INS PL Vehicle Policies

PA INS PL Vehicle Policies - Distribution Code

PA INS PL Vehicle Policies - Policy Status PC Code

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

Items

PA INS PL Vehicle Policies.As of Date

PA INS PL Vehicle Policies.Record Count

PA INS PL Vehicle Policies - Distribution Code.Distribution Code Description

PA INS PL Vehicle Policies - Policy Status PC Code.Policy Status PC Code Description

PA INS PL Vehicle Policies.Policy Term

PA INS PL Vehicle Policies.Inception Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty or Life insurance products (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written

Percentages

The Percent Distribution calculates the percentages for Record Count for each change in Org_Unit ID.

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Policies

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Policies

PA INS PL Vehicle Policies - Distribution Code
> PA INS PL Vehicle Policies

PA INS PL Vehicle Policies - Policy Status PC Code
> PA INS PL Vehicle Policies

13.8 Property and Casualty Insurance Profitability Report

The Property and Casualty Insurance Profitability Report shows accident year combined ratios (a common measure of insurance company profitability) at a product and coverage level.

Workbook

pa_ins_pc_profit.dis

Preparing to Run the Report

Each component used in computing the combined ratio is displayed in a separate worksheet. Because each worksheet is dependent on hierarchies, be sure to do the following:

1. Create three Tree Rollup IDs (one each for Common COA, Org Unit, and Coverage) in Oracle Performance Analyzer.

2. Transform the Tree Rollup IDs using the Transformation ID.

3. Name the transformation results, PA_HIER_PRODUCT, PA_HIER_ORG, and PA_HIER_COV, respectively.

Worksheets

Expense Ratio

ULAE Ratio

Case Reserves

Paid Loss and ALAE

IBNR Reserves

Combined Ratio

13.8.1 The Expense Ratio Worksheet

The Expense Ratio Worksheet has the following components:

Folders

PA INS PL Vehicle Coverages

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS PL Vehicle Coverages.Commissions

PA INS PL Vehicle Coverages.General Expense

PA INS PL Vehicle Coverages.Other Acquisition Costs

PA INS PL Vehicle Coverages.Premium Written Direct

PA INS PL Vehicle Coverages.Taxes License Fees

PA INS PL Vehicle Coverages.As of Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Total Expenses	General Expense SUM + Other Acquisition Costs SUM + Taxes License Fees SUM + Commissions SUM	Sum of various underwriting expenses
Expense Ratio	DECODE(Premium Written Direct, 0, 0, Total Expenses/Premium Written Direct SUM)	Divides the total underwriting expenses by the total direct written premium

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Coverages

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Coverages

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS PL Vehicle Coverages

13.8.2 ULAE Ratio Worksheet

The ULAE Ratio Worksheet has the following components:

Folders

PA INS PL Vehicle Coverages

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS PL Vehicle Coverages.Premium Earned Direct

PA INS PL Vehicle Coverages.Unallocated Loss Adj Expense

PA INS PL Vehicle Coverages.As of Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
ULAE Ratio	DECODE(Premium Earned Direct SUM,0,0,Unallocated Loss Adj Expense SUM/Premium Earned Direct SUM)	Divides total unallocated loss adjustment expenses by the direct earned premium.

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Coverages

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Coverages

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS PL Vehicle Coverages

13.8.3 Case Reserves Worksheet

The Case Reserves Worksheet has the following components:

Folders

PA INS PL Vehicle Claims

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS PL Vehicle Claims.As of Date

PA INS PL Vehicle Claims.ALAE Case Reserve Ending

PA INS PL Vehicle Claims.Loss Case Reserve Ending

PA INS PL Vehicle Claims. Loss Case Reserve EB

PA INS PL Vehicle Claims. ALAE Case Reserve EB

PA INS PL Vehicle Claims.Loss Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Loss and ALAE Case Reserves	Loss Case Reserve Ending SUM + Loss Case Reserve EB SUM + ALAE Case Reserve Ending SUM + ALAE Case Reserve EB SUM	Adds adjuster case reserves with bulk reserves for reported claims

Joins
PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Claims

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Claims

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS PL Vehicle Claims

Conditions
The As-of Date is the latest evaluation month, which you define.

13.8.4 Paid Loss and ALAE Worksheet

The Paid Loss and ALAE Worksheet has the following components:

Folders
PA INS PL Vehicle Claims

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items
PA INS PL Vehicle Claims.ALAE Amount Paid

PA INS PL Vehicle Claims.Loss Amount Paid

PA INS PL Vehicle Claims.Loss Date Year

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Paid Loss and ALAE	SUM_Loss Amount Paid + SUM_ ALAE Amount Paid	Cumulates paid losses and ALAE by loss date year

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS PL Vehicle Claims

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS PL Vehicle Claims

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS PL Vehicle Claims

13.8.5 IBNR Reserves Worksheet

IBNR reserves are ending balance entries stored in the Ledger Stat table. Each IBNR reserve has an associated organizational unit, product, coverage, and accident year. The financial element *leaf value = 100* identifies the reserve as an ending balance. You create GL Account leaf values to identify the accident year of the reserve.

The IBNR Reserves Worksheet looks for the following items:

- A transformed Ledger Stat table called PA_INS_OFS_RPT_INS_AY that contains the IBNR reserves. You apply a filter in the Transformation ID to capture only those GL Accounts that represent IBNR reserves.

- A GL hierarchy called PA_HIER_GL: PRIMARY GL HIER

Folders

PA INS OFS RPT INS AY

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

PA_HIER_GL: PRIMARY GL HIER

Items

PA INS OFS RPT INS AY.End Balance

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

PA_HIER_GL: PRIMARY GL HIER.GL Account ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (for example, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product.

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS OFS RPT INS AY

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS OFS RPT INS AY

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS OFS RPT INS AY

PA_HIER_GL: PRIMARY GL HIER (GL_ACCOUNT_ID)
> PA INS OFS RPT INS AY

13.8.6 Combined Ratio Worksheet

The Combined Ratio Worksheet summarizes the results of the preceding five worksheets. Those worksheets rely on the PA INS PL Vehicle Coverages, PA INS PL Vehicle Claims, and PA INS OFS RPT INS AY tables, so a view is defined to combine the results.

SQL Commands

The following SQL commands create the view:

```
create or replace view ins_profit_loss_total
    (loss_date_year,
    coverage_id,
    org_unit_id,
    common_coa_id,
    paid_loss_ALAE,
    Loss_ALAE_case_resevers,
    IBNR_Loss_reserves,
    direct_earned_premium,
    unallocated_loss_adj_exp,
    direct_written_premium,
    total_expenses)
as
(select to_char(loss_date,'YYYY'),
    coverage_id,
    org_unit_id,
    common_coa_id,
    sum(loss_amt_paid+alae_amt_paid),
    sum(decode(as_of_date,(select max(as_of_date) from pl_vehicle_claims),
loss_case_reserve_end+loss_case_reserve_end_bulk+alae_case_reserve_end+alae_
case_reserve_end_bulk,0)),
    0,
    0,
    0,
    0,
    0
    from pl_vehicle_claims
    group by
to_char(loss_date,'YYYY'),coverage_id,org_unit_id,common_coa_id
union
select to_char(end_of_year_date,'YYYY'),
    coverage_id,
    org_unit_id,
    common_coa_id,
    0,
    0,
    sum(end_bal),
    0,
    0,
    0,
    0
from ins_gl_years g, ofs_rpt_ins_ay l
```

```
    where g.gl_account_id=l.gl_account_id
    group by
to_char(end_of_year_date,'YYYY'),coverage_id,org_unit_id,common_coa_id
union
select to_char(as_of_date,'YYYY'),
    coverage_id,
    org_unit_id,
    common_coa_id,
    0,
    0,
    0,
    sum(prem_earned_direct),
    sum(unalloc_loss_adj_exp),
    sum(prem_written_direct),
sum(general_expense)+sum(other_acquisition_costs)+sum(taxes_license_
fees)+sum(commissions)
    from pl_vehicle_coverages
    group by
to_char(as_of_date,'YYYY'),coverage_id,org_unit_id,common_coa_id);
```

Folders

PA INS Ins Profit Loss Total

PA_HIER_PRODUCT: PRODUCT HIER

PA_HIER_ORG: PRIMARY HIER

PA_HIER_COV: COVERAGE HIER

Items

PA INS Ins Profit Loss Total.Loss Date Year

PA INS Ins Profit Loss Total.Paid Loss ALAE

PA INS Ins Profit Loss Total.Loss ALAE Case Reserves

PA INS Ins Profit Loss Total.IBNR Loss Reserves

PA INS Ins Profit Loss Total.Direct Earned Premium

PA_HIER_PRODUCT: PRODUCT HIER.Common COA ID Leaf Description

PA_HIER_ORG: PRIMARY HIER.Org Unit ID Leaf Description

PA_HIER_COV: COVERAGE HIER.Coverage ID Leaf Description

Page Items

Folder	Item	Description
PA_HIER_PRODUCT: PRODUCT HIER	Common COA ID Leaf Description	Identifies the policy form of the Property/Casualty insurance product (that is, each product is typically available in a variety of forms)
PA_HIER_ORG: PRIMARY HIER	Org Unit ID Leaf Description	The State and Company in which the policy is written
PA_HIER_COV: COVERAGE HIER	Coverage ID Leaf Description	Identifies the coverage, option, or rider associated with the insurance product

Calculations

Item	Formula	Description
Loss and ALAE Ratio	DECODE(Direct Earned Premium SUM,0,0,(Paid Loss Alae SUM+Loss Alae Case Reserves SUM+Ibnr Loss Reserves SUM)/Direct Earned Premium SUM)	Paid, case, and IBNR losses are combined and divided by the sum of direct earned premiums.
ULAE Ratio Summary	DECODE(Direct Earned Premium SUM,0,0,Unallocated Loss Adj Exp SUM/Direct Earned Premium SUM)	This is the ULAE ratio computed on the ULAE Ratio worksheet.
Expense Ratio Sum	DECODE(Direct Written Premium SUM,0,0,Total Expenses SUM/Direct Written Premium SUM)	This is the expense ratio computed on Expense Ratio worksheet.
Combined Ratio	Loss and ALAE Ratio + ULAE Ratio Summary + Expense Ratio Sum	Loss ratio is combined with the expense ratio to arrive at the combined ratio

Joins

PA_HIER_PRODUCT: PRODUCT HIER (COMMON_COA_ID)
> PA INS OFS RPT INS AY

PA_HIER_ORG: PRIMARY HIER (ORG_UNIT_ID)
> PA INS OFS RPT INS AY

PA_HIER_COV: COVERAGE HIER (COVERAGE_ID)
> PA INS OFS RPT INS AY

14

Oracle FDM Rate Manager Reports

Rate Manager reports provide interest rate and exchange rate information for scenario forecasting exercises. The reports also support multicurrency reporting within FDM, including the current European Monetary Union structure and the Euro.

Currently, only one report is available for the Rate Manager application: The Exchange Rate Report. This report shows the exchange rate matrix between currencies for multicurrency reporting.

Report

Exchange Rate Report

Workbook

exchng.dis

Folder

RTM Exchange Rate Hist

Items

RTM Exchange Rate Hist.From Currency Cd

RTM Exchange Rate Hist.To Currency Cd

RTM Exchange Rate Hist.Effective Date

RTM Exchange Rate Hist.Exchange Rate

Crosstab Layout

Vertical Axis = To Currency Cd

Horizontal Axis = From Currency Cd

Data Points = Exchange Rate

15

Using the Oracle Risk Manager Reports

Using both deterministic and stochastic methods, Oracle Risk Manager produces insightful value-added information for profitability scenario-based processing results and modern financial models to forecast various financial elements. Additionally, the reports support consolidated currency reporting with exchange rate gains and losses. This support enables multinational corporations to manage their exchange rate risk in the volatile global environment.

This chapter provides detailed information on the following reports:

15.1 Detail Cashflow Runoff Report

This report shows the detailed cashflow runoff by account. It is used to report the transactions created by the cashflow engine for the month.

Workbook

detcfrun.dis

Folders

FDM Process Cash Flows

FDM Product Hierarchy

Items

FDM Process Cash Flows.Id Number

FDM Process Cash Flows.Cash Flow Date

FDM Process Cash Flows.Result Sys Id

FDM Product Hierarchy.Leaf Description

Calculations

Item	Formula	Description
Beginning Balance	DECODE(Financial Elem Id,60,Float Value,0)	Returns values of Beginning Balance.
Ending Balance	DECODE(Financial Elem Id, 100, Float Value, 0)	Returns values of Ending Balance.
Interest Cash Flow	DECODE(Financial Elem Id, 430,Float Value, 0)	Returns values of Interest Cash Flow.
Payment Runoff - Positive	DECODE(Financial Elem Id,190,Float Value,0)	Returns values of Payment Runoff - Positive.
Repricing Balance	DECODE(Financial Elem Id, 250,Float Value,0)	Returns values of Repricing Balance.
Total Payment Amount	"Total Runoff-Positive" + Interest Cash Flow	Returns values of Total Payment Amount.
Total Runoff - Positive	DECODE (Financial elem Id,210,Float Value,0)	Returns values of Total Runoff - Positive.

Filter

Cash Flow Cd <> 32 (for example, initial interest for advance)

Join

FDM Process Cash Flows > FDM Product Hierarchy

15.2 Earnings at Risk Density Report

This report shows earnings at risk by product stratified into ten (10) tiers.

Workbook

eardensi.dis

Folder

Std EAR_LEAF_DTL_100882

Items

Std EAR_LEAF_DTL_100882.Bucket End Date

Std EAR_LEAF_DTL_100882.Bucket Start Date

Std EAR_LEAF_DTL_100882.Product Leaf

Calculations

Item	Formula	Description
EAR Density	DECODE(LEAST(GREATEST(Earnings,:T1L),:T1H),Earnings, '01. ' | | :T1L | | ' - ' | | :T1H	This decode statement looks for the lowest rate entered by the user (designated by :T1L, a parameter,) and for the greatest number (designated by :T1H, another parameter). The numbers reference the Earnings item and then print to screen the entered numbers. By replacing the Earnings with another item, the user can query on an item in the database. For example, replace it with Current Book Balance, and the user can specify a range of balances for a report.

15.3 Earnings at Risk Detail Report

This report shows the earnings at risk at the detail level by product and scenario with statistical calculations.

Workbook

eardtl.dis

Folders

Std Ear Leaf Dtl 102134

Items

Std Ear Leaf Dtl 102134.Earnings SUM

Std Ear Leaf Dtl 102134.End Date

Std Ear Leaf Dtl 102134.Leaf Node

Std Ear Leaf Dtl 102134.Rate Path Num

Std Ear Leaf Dtl 102134.Start Date

15.4 Earnings at Risk Summary Report

This report shows the earnings at risk at the detail level by income type and scenario with statistical calculations.

Workbook

earsum.dis

Folder

Std Ear Total Dtl 102134

Items

Std Ear Total Dtl 102134.End Date

Std Ear Total Dtl 102134.Rate Path Num

Std Ear Total Dtl 102134.Start Date

Calculations

Item	Formula	Description
EAR Income	DECODE(:Income Type,1,Net Interest Income,Net Income)	Returns the values of the income type specified by the user.
Selected Income Type	DECODE(:Income Type,1,'Net Interest Income','Net Income')	Shows the income type specified by the user for reporting purposes.

User Prompt
Income Type (1=Net Interest Income, 2=Net Income)

15.5 Income vs. Rate Report

This report compares short rates against earnings by product.

Workbook
inc-rate.dis

Folders
Std Ear Leaf Dtl 102134

Std Stochastic Stoch Assump

Items
Std Ear Leaf Dtl 102134.Earnings

Std Ear Leaf Dtl 102134.Leaf Node

Std Ear Leaf Dtl 102134.Start Date

Std Stochastic Stoch Assump.Tm Process Sys Id

Calculation

Item	Formula	Description
Rate Percent	Rate/100	Returns rates as percentage.

Filters

Calc Source Cd = 1

Tm Process Sys Id = 102134

Joins

Std Ear Leaf Dtl 102134 -> Std Stochastic Interest Rates Audit

Std Stochastic Interest Rates Audit -> Std Stochastic Stoch Assump

15.6 Detail/Summary Cashflow Report

This report shows detailed information about payment, prepayment, and repricing activity.

Workbook

rmcashfl.dis

Folders

CCOA ID Desc

Std Ofs Rpt Res Dtl 102275 C

Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code

Std Ofs Rpt Res Dtl 102275 C - Scenario Description

Items

Table 15–1   Detail/Summary Cashflow Report Items

Folder	Item	Description
CCOA ID Desc	Common COA ID Desc	Shows the COA ID description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 C	Average Bal	Shows the average balance for the ending period.
Std Ofs Rpt Res Dtl 102275 C	Beginning Balance	Shows the beginning balance for the period.
Std Ofs Rpt Res Dtl 102275 C	Deferred Runoff	Shows deferred runoff as the result of this RM process.

Table 15–1   (Cont.) Detail/Summary Cashflow Report Items

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 C	End Balance	Shows the end balance for the ending period.
Std Ofs Rpt Res Dtl 102275 C	End Date	Shows the end date for each period.
Std Ofs Rpt Res Dtl 102275 C	Interest Accrued	Shows the interest accrued amount at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Interest Cash Flow	Shows the interest cash flow at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Interest Credited	Shows the interest credited for this period.
Std Ofs Rpt Res Dtl 102275 C	Maturity Runoff-Negative	Shows negative maturity runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Maturity Runoff-Positive	Shows positive maturity runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	New Add Balance	Shows the new business balance at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Payment Runoff-Negative	Shows negative payment runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Payment Runoff-Positive	Shows positive payment runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Repricing Balance	Shows the repricing balance for the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Roll Add Balance	Shows the balance from rolled over monies at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Start Date	Shows the start date for each period.
Std Ofs Rpt Res Dtl 102275 C	Total Runoff-Negative	Shows negative runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Total Runoff-Positive	Shows positive runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted After Repricing Net Rate	Shows Weighted After Repricing Net Rate as the result of this RM process.

Table 15–1 (Cont.) Detail/Summary Cashflow Report Items

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 C	Weighted Average Net Rate	Shows Weighted Average Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Before Repricing Net Rate	Shows Weighted Before Repricing Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Beginning Net Rate	Shows Weighted Beginning Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Ending Net Rate	Shows Weighted Ending Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Fully Indexed Net Rate	Shows Weighted Fully Indexed Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted New Add Net Rate	Shows Weighted New Add Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Roll Add Net Rate	Shows Weighted Roll Add Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted Total Runoff Net Rate	Shows Weighted Total Runoff Net Rate as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Weighted WARM	Shows Weighted WARM as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code	ISO Currency Code Description	Shows the ISO Currency description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 C - Scenario Description	Result Scenario Description	Shows the scenario description for each one of the processes.

Calculations

Table 15–2   Detail/Summary Cashflow Report Calculations

Item	Formula	Description
ACCRUAL INCOME	SUM(NULL)	A label for Accrual Income
Accumulated Translation Balance	SUM(Accumulated Translation Amount)/1000	Divides by 1000 for reporting purposes.
Ave Balance	SUM(Average Bal)/1000	Divides Average Bal by 1000 for reporting purposes.
Beginning Balance	SUM(Beginning Balance)/1000	Divides Beginning Balance by 1000 for reporting purposes.
BEGINNING BALANCE	SUM(NULL)	A label for Beginning Balances.
CASH FLOW ACTIVITY	SUM(NULL)	A label for Cash Flow Activity.
CURRENCY ACTIVITY	SUM(NULL)	A label for Currency Activity.
Deferred-Runoff	SUM(Deferred Runoff)/1000	Divides Deferred Runoff by 1000 for reporting purposes.
Ending Balance	SUM(End Balance)/1000	Divides End Balance by 1000 for reporting purposes.
ENDING BALANCES	SUM(NULL)	A label for Ending Balance.
Interest Credited	SUM(Interest Credited)/1000	Divides Interest Credited by 1000 for reporting purposes.
Interest-Accrued	SUM(Interest Accrued)/1000	Divides Interest Accrued by 1000 for reporting purposes.
Interest-Cash Flow	SUM(Interest Cash Flow)/1000	Divides Interest Cash Flow by 1000 for reporting purposes.
NEW BUSINESS ACTIVITY	SUM(NULL)	A label for New Business Activity.
Originations	SUM(New Add Balance)/1000	Divides New Add Balance by 1000 for reporting purposes.
Prepayment	Sum("Prepay Runoff-Positive + Prepay Runoff Neg)/1000	Nets prepayments and divides by 1000 for reporting purposes.

Table 15–2   (Cont.) Detail/Summary Cashflow Report Calculations

Item	Formula	Description
Principal at Maturity	SUM("Maturity Runoff - Negative"+"Maturity Runoff - Positive")/1000	Calculates Total Principal at Maturity by summing negative and positive and dividing the sum by 1000 for reporting purposes.
Realized Currency Gain/Loss (Interest)	SUM("Realized Currency Gain/Loss (Interest)")/1000	Divides Realized Currency Gain/Loss (Interest) by 1000 for reporting purposes.
Realized Currency Gain/Loss (Principal)	SUM("Realized Currency Gain/Loss (Principal)")/1000	Divides Realized Currency Gain/Loss (Principal) by 1000 for reporting purposes.
Reprice Balance	SUM(Repricing Balance)/1000	Divides Repricing Balance by 1000 for reporting purposes.
REPRICING ACTIVITY	Sum(Null)	A label for Repricing Activity.
Rolling Balances	SUM(Roll Add Balance)/1000	Divides Roll Add Balance by 1000 for reporting purposes.
Scheduled Principal Runoff	SUM("Payment Runoff - Negative"+"Payment Runoff - Positive")/1000	Calculates Total Principal Runoff by summing negative and positive and dividing the sum by 1000 for reporting purposes.
Std Ofs Rpt Res Dtl 102275 C	Prepay Runoff - Neg	Shows the negative Prepayment runoff at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Prepay Runoff - Pos	Shows the positive Prepayment runoff at the end of the period.
Total Cash Flow	Total Principal Runoff+SUM(Interest Cash Flow-Interest Credited)/1000	Calculates Total Cash Flow by summing the Total principal Runoff calculation with interest cash flow and interest credited. Then the sum is divided by 1000 for reporting purposes.
Total Currency Gain/Loss (Principal)	SUM("Total Currency Gain/Loss (Principal)")/1000	Divides Total Currency Gain/Loss (Principal) by 1000 for reporting purposes.
Total New Business	SUM(New Add Balance+Roll Add Balance)/1000	Calculates Total New Business by summing New Add Balance and Roll Add Balance and dividing by 1000 for reporting purposes.

Table 15–2   (Cont.) Detail/Summary Cashflow Report Calculations

Item	Formula	Description
Total Principal Runoff	SUM("Total Runoff - Negative"+"Total Runoff - Positive")/1000	Calculates Total Principal Runoff by summing negative and positive and dividing the sum by 1000 for reporting purposes.

Joins

CCOA ID Desc > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - Scenario Description > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - ISO Currency Code > Ofs Rpt Res Dtl 102275 C

15.7 Consolidated Detail / Summary Cashflow Report

This report runs against the consolidated RM result detail table (such as Std Ofs Rpt Res Dtl 102275$C), providing the same functionality as its unconsolidated report counterpart. Since these results are generated using future scenario and exchange rate assumptions, conversion using historical rates is inappropriate. The Risk Manager processing engine allows users to consolidate their result details into a single user-specified functional currency. See the *Oracle Financial Data Manager Rate Manager Reference Guide* for more information.

Workbook

rmcashfl_mc.dis

15.8 Gap Detail Report

This report shows account level gap runoff information for a single start date.

Workbook

rmgapdet.dis

Folders

Std Ofs Rpt Res Dtl 102275 G

Std Ofs Rpt Res Dtl 102275 G - ISO Currency Code

Std Ofs Rpt Res Dtl 102275 G - Scenario Description

Std Rpt Result Master

Items

Table 15–3   Gap Detail Report Items

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 G	Dynamic Gap Date	This is the as of date for the gap process.
Std Ofs Rpt Res Dtl 102275 G	End Date	The end date for the gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Accrued Interest Net	The net accrued interest for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Deferred Runoff	The deferred runoff amount for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Interest Cash Flow Net	The net interest cash flow for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Interest Credited	The interest credited for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Principal Runoff	The principal runoff amount for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Repricing Runoff	The repricing runoff amount for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Runoff	The runoff amount for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Gap Runoff Net Rate	Weighted Gap rate divided by gap balance.
Std Ofs Rpt Res Dtl 102275 G	Gap Runoff Term	The period covered for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Start Date	The start date for the gap process.
Std Ofs Rpt Res Dtl 102275 G - ISO Currency Code	ISO Currency Code Description	Shows the ISO Currency description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 G - Scenario Description	Std Rpt Res Dtl Gap Scenario Description.Result Scenario Description	Describes the scenario for the gap process.

Table 15–3   (Cont.) Gap Detail Report Items

Folder	Item	Description
Std Rpt Result Master	Financial Rollup Code	Necessary for Gap to calculate the correct signage

Calculations

Item	Formula	Description
Gap Accrued Interest Net	SUM(DECODE(Financial Rollup Code,300,Gap Accrued Interest Net*-1,310,Gap Accrued Interest Net*-1,350,Gap Accrued Interest Net*-1,375,Gap Accrued Interest Net*-1,400,Gap Accrued Interest Net*-1,500,Gap Accrued Interest Net*-1,600,Gap Accrued Interest Net*-1,620,Gap Accrued Interest Net*-1,Gap Accrued Interest Net))	Returns the correctly signed values for gap.
Interest Gap Runoff	SUM(NULL)	A label for Interest Gap Runoff.
Principal Gap Runoff	SUM(NULL)	A label for Principal Gap Runoff.
Runoff Life	DECODE(SUM(Gap Runoff), 0, 0, SUM(Gap Runoff Term) / SUM(Gap Runoff))	Weights Gap Runoff Term by Gap Runoff.

Joins

Ofs Rpt Res Dtl 102275 G - Scenario Description > Ofs Rpt Res Dtl 102275 G

Ofs Rpt Res Dtl 102275 G - ISO Currency Code > Ofs Rpt Res Dtl 102275 G

Std Rpt Result Master -> Std Ofs Rpt Res Dtl 102275 G

15.9 Consolidated Gap Detail Report

This report runs against the consolidated RM result detail table (such as Std Ofs Rpt Res Dtl 102275$G), providing the same functionality as its unconsolidated report counterpart. Since these results are generated using future scenario and exchange rate assumptions, conversion using historical rates is inappropriate. The Risk Manager processing engine allows users to consolidate their result details into a single user-specified functional currency.

See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features," for more information.

Workbook

rmgapdet_mc.dis

15.10 Gap Summary Report

This report shows both balance and off-balance sheet gap positions for a single start date.

Workbook

rmgapsum.dis

Folders

Ofs Rpt Hier Common Coa Id

Std Ofs Rpt Res Dtl 102275 G

Std Ofs Rpt Res Dtl 102275 G - ISO Currency Code

Std Ofs Rpt Res Dtl 102275 G - Scenario Description

Items

Table 15–4   Gap Summary Report

Folder	Item	Description
Ofs Rpt Hier Common Coa Id	Level 05 Desc	Level 05 description is the next level down from level 06 that will return items such as earning asset and total liabilities.
Ofs Rpt Hier Common Coa Id	Level 06 Desc	Level 06 will return the assets, liabilities, and capital for this gap process.
Ofs Rpt Hier Common Coa Id	Level 07 Desc	This is the highest level of the rollup.
Std Ofs Rpt Res Dtl 102275 G	Dynamic Gap Date	This is the as of date for the gap process.
Std Ofs Rpt Res Dtl 102275 G	End Date	The end date for the gap process.

Table 15–4   Gap Summary Report

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 G	Gap Runoff	The runoff amount for this gap process.
Std Ofs Rpt Res Dtl 102275 G	Start Date	The start date for the gap process.
Std Ofs Rpt Res Dtl 102275 G - ISO Currency Code	ISO Currency Code Description	Shows the ISO Currency description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 G - Scenario Description	Result Scenario Description	Describes the scenario for the gap process.

Filters

"Level 06 Desc" = 'Total Assets'

"Level 06 Desc" IN ('Total Liabilities & Capital','Total Assets')

"Level 07 Desc" = 'Total Rollup'

Joins

Ofs Rpt Hier Common Coa Id -> Ofs Rpt Res Dtl 102275 G

Ofs Rpt Res Dtl 102275 G - Scenario Description -> Ofs Rpt Res Dtl 102275 G

Ofs Rpt Res Dtl 102275 G - ISO Currency Code -> Ofs Rpt Res Dtl 102275 G

15.11 Consolidated Gap Summary Report

This report runs against the consolidated RM result detail table (such as Std Ofs Rpt Res Dtl 102275$G), providing the same functionality as its unconsolidated report counterpart. Since these results are generated using future scenario and exchange rate assumptions, conversion using historical rates is inappropriate. The Risk Manager processing engine allows users to consolidate their result details into a single user-specified functional currency. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features," for more information.

Workbook

rmgapsum_mc.dis

15.12 Scenario Income Statement

This report shows the change in interest income for a multitude of time periods at both the product and scenario level.

Workbook

rmincsce.dis

Folders

Ofs Rpt Hier Common Coa Id

Std Ofs Rpt Res Dtl 102275 C

Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code

Std Ofs Rpt Res Dtl 102275 C - Result Type Description

Std Ofs Rpt Res Dtl 102275 C - Scenario Description

Std Rpt Result Master

Items

Table 15–5 Scenario Income Statement Items

Folder	Item	Description
Ofs Rpt Hier Common Coa Id	Leaf Description	Shows the product level description.
Std Ofs Rpt Res Dtl 102275 C	Deferred Runoff	Shows deferred runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Dividends	Shows dividends declared as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	End Date	End date is the ending date for the month.
Std Ofs Rpt Res Dtl 102275 C	Federal Taxes	Shows the federal taxes as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Interest Accrued	Shows the interest accrued amount at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Interest Cash Flow	Shows the interest cash flow at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Local Taxes	Shows the local taxes as the result of this RM process.

Table 15–5   (Cont.) Scenario Income Statement Items

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 C	Non Interest Expense	This item contains the non-interest expense information for the given period.
Std Ofs Rpt Res Dtl 102275 C	Non Interest Income	This item contains the non-interest income information for the given period.
Std Ofs Rpt Res Dtl 102275 C	Start Date	Shows the start date for each period.
Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code	ISO Currency Code Description	Shows the ISO Currency description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 C - Result Type Description	Result Type	Shows the result type description.
Std Ofs Rpt Res Dtl 102275 C - Scenario Description	Result Scenario Description	Shows the scenario description for each one of the processes.
Std Rpt Result Master	Financial Rollup Code	Necessary to calculate correct signage

Calculations

Item	Formula	Description
Int Accrued	SUM(DECODE(Financial Rollup Code,300,Interest Accrued*-1,310,Interest Accrued*-1,350,Interest Accrued*-1,375,Interest Accrued*-1,400,Interest Accrued*-1,500,Interest Accrued*-1,600,Interest Accrued*-1,620,Interest Accrued*-1,Interest Accrued))+SUM(Non Interest Income-Non Interest Expense+Federal Taxes+Local Taxes+Dividends)	Adds all income and subtracts expenses including taxes and dividends.

Joins

Ofs Rpt Hier Common Coa Id > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - ISO Currency Code > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - Result Type Description > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - Scenario Description > Ofs Rpt Res Dtl 102275 C

Std Rpt Result Master > Std Ofs Rpt Res Dtl 102275 C

15.13 Consolidated Scenario Income Statement

This report runs against the consolidated RM result detail table (such as Std Ofs Rpt Res Dtl 102275$C), providing the same functionality as its unconsolidated report counterpart. Since these results are generated using future scenario and exchange rate assumptions, conversion using historical rates is inappropriate. The Risk Manager processing engine allows users to consolidate their result details into a single user-specified functional currency. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features," for more information.

Workbook

rmincsce_mc.dis

15.14 Summary Income Statement

This report shows the income statement based on scenarios instead of current position.

Workbook

rmincsum.dis

Folders

Std Ofs Rpt Res Dtl 102275 C

Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code

Std Ofs Rpt Res Dtl 102275 C - Result Type Description

Std Ofs Rpt Res Dtl 102275 C - Scenario Description

Std Rpt Result Master

Items

Folder	Item	Description
Std Ofs Rpt Res Dtl 102275 C	Deferred Runoff	Shows deferred runoff as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Dividends	Shows dividends declared as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	End Date	End date is the ending date for the month.
Std Ofs Rpt Res Dtl 102275 C	Federal Taxes	Shows the federal taxes as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Interest Accrued	Shows the interest accrued amount at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Interest Cash Flow	Shows the interest cash flow at the end of the period.
Std Ofs Rpt Res Dtl 102275 C	Local Taxes	Shows the local taxes as the result of this RM process.
Std Ofs Rpt Res Dtl 102275 C	Non Interest Expense	This item contains the non-interest expense information for the given period.
Std Ofs Rpt Res Dtl 102275 C	Non Interest Income	This item contains the non-interest income information for the given period.
Std Ofs Rpt Res Dtl 102275 C	Start Date	Shows the start date for each period.
Std Ofs Rpt Res Dtl 102275 C - ISO Currency Code	ISO Currency Code Description	Shows the ISO Currency description in relationship to the Result Detail process.
Std Ofs Rpt Res Dtl 102275 C - Result Type Description	Result Type	Shows the result type description.
Std Ofs Rpt Res Dtl 102275 C - Scenario Description	Result Scenario Description	Shows the scenario description for each one of the processes.
Std Rpt Result Master	Financial Rollup Code	Necessary to calculate correct signage.

Calculations

Item	Formula	Description
Dividends	Sum(Dividends * -1) or 0	Returns Dividends *(see Note, following table)*
Federal Taxes	Sum(Federal Taxes * -1) or 0	Returns Federal Taxes*
Income after Taxes	Net Income before Taxes + Federal Taxes+Local Taxes	Returns Total Income After Taxes.
Interest Expense	Sum(Decode(Financial Rollup Code, 300,Interest Accrued *-1, 620,Interest Accrued *-1,0))	Returns Interest Expense.
Interest Income	Sum(decode(Financial Rollup Code, 100,Interest Accrued, 610, Interest Accrued,0)) + sum(Deferred Runoff)	Returns Interest Income.
Local Taxes	Sum(Local Taxes *-1)	Returns Local Taxes.
Net Income Before Taxes	Net Interest Income + Net Non-Interest Income	Sums Net Interest Income with Net Non Interest Income.
Net Interest Income	Off B/S Income + Interest Expense + Interest Income	Sums the interest income by adding Interest Accrued and Deferred Runoff.
Net Non-Interest Income	SUM(Non Interest Income - Non Interest Expense)	Subtracts Non Interest Expense from Non Interest Income.
Off B/S Income	SUM(Decode(Financial Rollup code, 110, Interest Accrued, 310, Interest Accrued *-1,0))	Returns Off Balance Sheet income items.

Note: Due to differences among foreign countries in dividend and tax regulations, Oracle Risk Manager only outputs the dividend and Federal tax data to the consolidated result detail tables (such as $C). Thus, this field is set to zero in the Summary Income Statement, but is multiplied by negative one in the Consolidated Summary Income Statement.

Joins

Ofs Rpt Res Dtl 102275 C - Scenario Description > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - Result Type Description > Ofs Rpt Res Dtl 102275 C

Ofs Rpt Res Dtl 102275 C - ISO Currency Code > Ofs Rpt Res Dtl 102275 C

Std Rpt Result Master > Std Ofs Rpt Res Dtl 102275 C

15.15 Consolidated Summary Income Statement

This report runs against the consolidated RM result detail table (such as Std Ofs Rpt Res Dtl 102275$C), providing the same functionality as its unconsolidated report counterpart. Since these results are generated using future scenario and exchange rate assumptions, conversion using historical rates is inappropriate. The Risk Manager processing engine allows users to consolidate their result details into a single user-specified functional currency. See "Registering PL/SQL Functions for Multicurrency" in Chapter 4, "Implementing Advanced FDM Reporting Features," for more information.

Workbook

rmincsum_mc.dis

15.16 Market Value Report

This report shows product level information about market values for a given start date.

Workbook

rmmktval.dis

Folders

Ofs Rpt Hier Common Coa Id

Std Rpt Result Bucket

Std Rpt Result Master

Std Rpt Result Scenario

Items

Folder	Item	Description
Ofs Rpt Hier Common Coa Id	Leaf Description	Shows the products names instead of product number.
Std Rpt Result Bucket	From Date 001	Shows the start date for a given RM process.
Std Rpt Result Master	Duration	Shows the price sensitivity of the market in relation to Market Value.
Std Rpt Result Master	Market Value	Shows the Market Value of a given product for each RM process.
Std Rpt Result Scenario	Result Scenario Description	Describes the scenario for the gap process.

Calculations

Item	Formula	Description
Duration	Decode(sum(Market Value),0,0,sum(Duration)/ Sum(Market Value)	Divides Duration by Market Value to return a weighted duration.
Start Date	"From Date 001"-1	This date comes from the first bucket in the result bucket table. Then one day is subtracted for the actual start date.
As of Date	"From Date 001"-1	This date comes from the first bucket in the result bucket table. Then one day is subtracted for the actual start date.

Joins

Std RptResult Scenario > Std Rpt Result Master

Ofs Rpt Hier Common Coa Id > Std Rpt Result Master

Std Rpt Result Bucket > Std Rpt Result Master

Special Considerations

The Market Value Report is unique in that it uses tables that cannot be registered through FDM/Discoverer Integrator. The administrator must register the OFSA Result Bucket and OFSA Result Scenario tables using Discoverer Administration Edition. For specifics on how to register a table from the database, see the *Oracle Discoverer 3.1 Administration Guide*.

> **Caution:**   When registering the OFSA Result Bucket table, always deselect Date Hierarchies from *Load Wizard: Step 4*. If the flag is left on, then the download will stop responding.

After the four tables are loaded (OFSA Result Master, OFSA Result Scenario, OFSA Result Bucket, and the Hierarchy Table), the joins need to be created.

Both the OFSA Result Scenario and OFSA Result Bucket need double joins to the OFSA Result Master table. Join by SYSTEM ID and Scenario Number, and ensure that the lookup tables (OFSA Result Scenario and OFSA Result Bucket) are used as the master folders.

After following the preceding steps, the user is ready to create the report.

15.17 Product, Rank, VaR Report

This report shows Product Description at page level, with Rate Path Number, Probability Ranking, and VaR as axis items.

Workbook

rmvar1.dis

Folders

Std Rpt Leaf Desc

Std Rpt VaR

Items

Std Rpt Leaf Desc.Description

Std Rpt VaR.Rate Path Number

Std Rpt VaR.Probability

Std Rpt VaR.Value at Risk

Join

Leaf Desc > Std Rpt VaR

Special Considerations

This report can be presented in a standard bar graph with the VaR numbers on the vertical axis and the Product on the horizontal axis. Select the graph option in the menu bar, and the graph appears. Any changes made to the data automatically change the graph. The data is multiplied by a negative one to return to numbers as positive for graphing purposes. The true power of this report lies in its graph and not the actual data report.

15.18 Rank, VaR (Total Bank Level) Report

This report shows Rate Path Number, Probability Ranking and VaR as axis items.

Workbook

rmvar2.dis

Folders

Std Rpt Leaf Desc

Std Rpt VaR

Items

Std Rpt VaR.Rate Path Number

Std Rpt VaR.Probability

Std Rpt VaR.Value at Risk

Special Considerations

This report can be presented in a standard bar graph with the VaR numbers on the vertical axis and the Product on the horizontal axis. Select the graph option in the menu bar, and the graph appears. Any changes made to the data automatically change the graph. The data is multiplied by a negative one to return to numbers as positive for graphing purposes. This true power of this report lies in its graph and not the actual data report.

15.19 Rank, Product, VaR Report

This report shows Probability Ranking at page level, with Product Description and VaR as axis items.

Workbook

rmvar3.dis

Folders

Std Rpt Leaf Desc

Std Rpt VaR

Items

Std Rpt Leaf Desc.Description

Std Rpt VaR.Probability

Calculation

Item	Formula	Description
VaR	Value at Risk*-1	This calculation multiplies the VaR number by negative one so that the numbers report as positive for graphing purposes.

Join

Leaf Desc > Std Rpt VaR

Special Considerations

This report can be presented in a standard bar graph with the VaR numbers on the vertical axis and the Product on the horizontal axis. Select the graph option in the menu bar, and the graph appears. Any changes made to the data automatically change the graph. The data is multiplied by a negative one to return to numbers as positive for graphing purposes. This true power of this report lies in its graph and not the actual data report.

15.20 Stochastic Market Value Report

This report shows the stochastic market values and total return by product.

Workbook

stochmkt.dis

Folders

Std Rpt Leaf Desc

Std Rpt Stochastic Market Value

Items

Std Rpt Leaf Desc.Description

Std Rpt Stochastic Market Value.TM Process ID

Std Rpt Stochastic Market Value.Current Balance SUM

Std Rpt Stochastic Market Value.Market Value SUM

Join

Std Rpt Leaf Desc > Std Rpt Stochastic Market Value

15.21 Stochastic Rates Report

This report shows the stochastic rates curve for a given bucket period.

Workbook

stochrate.dis

Folder

RM Stochastic Interest Rates Audit

Items

RM Stochastic Interest Rates Audit.Bucket Start Date

RM Stochastic Interest Rates Audit.Process Sys Id

RM Stochastic Interest Rates Audit.Rate

RM Stochastic Interest Rates Audit.Scenario Num

Calculations

Item	Formula	Description
INTEREST RATE	TO_CHAR(RM Stochastic Interest Rates Audit.Interest Rate Cd) | | ' - '| |Irc Name	Returns concatenated string joining Interest Rate Cd to Irc Name.

Filter

Calc Source Cd = 1

Join

RM Stochastic IRCS -> RM Stochastic Interest Rates Audit

15.22 Term Structure Parameters Report

This report shows the term structure parameters as of the effective date.

Workbook

termspar.dis

Folder

RM Stochastic Irc Ts Param Hist

Item

RM Stochastic Irc Ts Param Hist.Effective Date

Calculations

Table 15–6   Term Structure Parameters Report Calculations

Item	Formula	Description
INTEREST RATE	RM Stochastic Irc Ts Param Hist.Interest Rate Cd | |Irc Name	Returns concatenated string joining Interest Rate Cd to Irc Name
Long Run Rate	Long Run Rate SUM/100	Return Long Run Rate in percentage

Table 15–6   (Cont.) Term Structure Parameters Report Calculations

Item	Formula	Description
Speed of Mean Reversion	Speed of Mean Reversion SUM	Return Speed of Mean Reversion
Volatility Merton	Volatility Merton SUM/100	Return Volatility Merton in percentage
Volatility Vasicek	Volatility Vasicek SUM/100	Return Volatility Vasicek in percentage

Join

RM Stochastic IRCS > RM Stochastic Irc Ts Param Hist

15.23 FASB 133 Reports

The FASB 133 Reports provide accounting information according to the FASB Statement No. 133. These reports show hedged balance sheet instruments together with individual derivative instruments.

Workbook

fasb133.dis

Derivative Instruments

Forward Contracts

Interest Rate Options

Interest Rate Swaps

15.23.1 Forward Contracts

This report shows the accounting information for Forward Contract derivative instruments and hedging activities.

Folders

Std Forward Contracts

Std Rpt Instrument

Items

Std Forward Contracts.Id Number

Std Rpt Instrument.Id Number

Calculations

Item	Formula	Description
MARKET PREMIUM - FC	Premium	Returns premium of the forward contract derivative instrument.
MARKET VALUE - INSTRUMENT	Market Value Code*Current Gross Par Balance/100	Returns market value of the instrument.
MATURITY - FC	Settlement Date	Returns settlement date of the forward contract derivative instrument.
MATURITY - INSTRUMENT	Std Rpt Instrument.Maturity Date	Returns maturity date of the instrument.
NOTIONAL - FC	Notional Balance/1	Returns notional balance of the forward contract derivative instrument.
NOTIONAL - INSTRUMENT	Current Net Par Balance/1	Returns current net par balance of the instrument.
RATE - FC	Cur Net Rate/100	Returns current net rate of the forward contract derivative instrument.
RATE - INSTRUMENT	Current Net Rate/100	Returns current net rate of the instrument.
TYPE - FC	'FRA'	Returns the type of forward contract derivative instrument.

Join

Std Forward Contracts > Std Rpt Instrument

15.23.2 Interest Rate Options

This report shows the accounting information for Interest Rate Options derivative instruments and hedging activities.

Folders

Std Interest Rate Options

Std Rpt Instrument

Items

Std Rpt Instrument.Id Number

Std Interest Rate Options.Id Number

Calculations

Table 15–7   Interest Rate Options Calculations

Item	Formula	Description
MARKET PREMIUM - IRO	Premium	Returns premium of the interest rate option derivative instrument.
MARKET VALUE - INSTRUMENT	Market Value Code*Current Gross Par Balance/100	Returns market value of the instrument.
MATURITY - INSTRUMENT	Std Rpt Instrument.Maturity Date	Returns maturity date of the instrument.
MATURITY - IRO	Std Interest Rate Options.Maturity Date	Returns maturity date of the interest rate option derivative instrument.
NOTIONAL - INSTRUMENT	Current Net Par Balance/1	Returns current net par balance of the instrument.
NOTIONAL - IRO	Notional Balance/1	Returns notional balance of the interest rate option derivative instrument.
RATE - INSTRUMENT	Current Net Rate/100	Returns current net rate of the instrument.
RATE - IRO	Cur Net Rate/100	Returns current net rate of the interest rate option derivative instrument.

Table 15–7   (Cont.) Interest Rate Options Calculations

Item	Formula	Description
TYPE - IRO	DECODE(Rate Cap Type Cd,0,(DECODE(Rate Floor Type Cd,1,'FLOOR PURCHASED')),1,(DECODE(Rate Floor Type Cd,0,'CAP PURCHASED')),2,(DECODE(Rate Floor Type Cd,0,'CAP SOLD')),'INTEREST RATE OPTION')	Returns the type of interest rate option derivative instrument.

Join
Std Interest Rate Options > Std Rpt Instrument

15.23.3 Interest Rate Swaps

This report shows the accounting information for Interest Rate Swaps derivative instruments and hedging activities.

Folders
Std Interest Rate Swaps

Std Rpt Instrument

Items
Std Rpt Instrument.Id Number

Std Interest Rate Swaps.Id Number

Calculations

Table 15–8   Interest Rate Swaps Calculations

Item	Formula	Description
MARKET PREMIUM - IRS	Premium	Returns premium of the interest rate swap derivative instrument.
MARKET VALUE - INSTRUMENT	Market Value Code*Current Gross Par Balance/100	Returns market value of the instrument.

Table 15–8   (Cont.) Interest Rate Swaps Calculations

Item	Formula	Description
MATURITY - INSTRUMENT	Std Rpt Instrument.Maturity Date	Returns maturity date of the instrument.
MATURITY - IRS	Std Interest Rate Swaps.Maturity Date	Returns maturity date of the interest rate swap derivative instrument.
NOTIONAL - INSTRUMENT	Current Net Par Balance/1	Returns current net par balance of the instrument.
NOTIONAL - IRS	Notional Balance/1	Returns notional balance of the interest rate swap derivative instrument.
RATE - INSTRUMENT	Current Net Rate/100	Returns current net rate of the instrument.
RATE - IRS	Cur Net Rate/100	Returns current net rate of the interest rate swap derivative instrument.
TYPE - IRS	DECODE(Rcv Adjustable Type Cd,0,DECODE(Pay Adjustable Type Cd,1,'SWAP RECEIVE FIXED'),1,DECODE(Pay Adjustable Type Cd,0,'SWAP PAY FIXED'),'INTEREST RATE SWAP')	Returns the type of interest rate swap derivative instrument.

Join

Std Interest Rate Swaps > Std Rpt Instrument

16

Oracle Transfer Pricing Reports

The Oracle Transfer Pricing reports track the profitability of different operations and product lines by reporting the transfer rate on financial products. Using the same sophisticated stochastic techniques available in Oracle Risk Manager, the Transfer Pricing processing engine calculates and outputs the cost of optionality. Based on the spread over transfer rate, this calculation quantifies the cost of optionality for a single instrument at the time of funding, thus isolating the line officers from the variability of funding costs.

This chapter provides detailed information on the following reports:

- Transfer Pricing Income Statement
- Transfer Pricing Option Cost Instrument Report

16.1 Transfer Pricing Income Statement

This report shows the forecasted contribution to the funding center by assets and liabilities. It also reports the net income generated by the funding center by month and scenario.

Workbook

rminctp.dis

Folders

Reporting Res Dtl C: REPORTING

Reporting Res Dtl C: REPORTING - Result Type Description

Reporting Res Dtl C: REPORTING - Scenario Description

Std Rpt Result Master

Items

Reporting Res Dtl C: REPORTING.Charge/Credit

Reporting Res Dtl C: REPORTING.End Date

Reporting Res Dtl C: REPORTING.Interest Accrued

Reporting Res Dtl C: REPORTING - Result Type Description.Result Type

Reporting Res Dtl C: REPORTING - Scenario Description.Result Scenario Description

Reporting Res Dtl C: REPORTING.Start Date

Std Rpt Result Master.Financial Rollup Code

Calculations

Table 16–1   Transfer Pricing Income Statement Calculations

Item Name	Formula	Description
Assets	SUM(NULL)	Label
Charge for funding	SUM(DECODE(Financial Rollup Code,100,"Charge/Credit",110,"Charge/Credit",610,"Charge/Credit",0))	Charge for funding collects the assets from Charge/ Credit financial element.
Credit on funding	SUM(DECODE(Financial Rollup Code,300,"Charge/Credit",310,"Charge/Credit",620,"Charge/Credit",0))	Credit on funding collects the liabilities from Charge/ Credit financial element.
Expense from Funding Center	Credit on funding	Total Expense for Funding Center.
Funding Center	SUM(NULL)	Label
Funding Center Net Income	Income to Funding Center - Expense from Funding Center	Total Net Income for the Funding Center.
Income to Funding Center	Charge for funding	Total Income for Funding Center.
Interest Earned on assets	SUM(DECODE(Financial Rollup Code,100,Interest Accrued,110,Interest Accrued,610,Interest Accrued,0))	Interest Earned on assets collects the assets from the Interest Accrued financial element.
Interest Expense	Sum(DECODE(Financial Rollup Code, 300,Interest Accrued,310,Interest Accrued,620,Interest Accrued,0))	Interest Expense collects the expenses from the Interest Accrued financial element.

Table 16–1   (Cont.) Transfer Pricing Income Statement Calculations

Item Name	Formula	Description
Liabilities	SUM(NULL)	Label
Net Assets	Interest Earned on assets - Charge for funding	Total Net Assets
Net Liabilities	Credit on funding - Interest Expense	Net of Interest expense from Credit on funding.

Joins

Reporting Res Dtl C: REPORTING - Result Type Description
> Reporting Res Dtl C: REPORTING

Std Rpt Result Master
> Reporting Res Dtl C: REPORTING

Reporting Res Dtl C: REPORTING - Scenario Description
> Reporting Res Dtl C: REPORTING

16.2 Transfer Pricing Option Cost Instrument Report

This report shows the Transfer Pricing option cost rate by instrument, product, and common COA.

Workbook

tpocinstr.dis

Folders

CCOA ID Desc

Mortgages1 - Product Type Code

Mortgages1 - Instrument Type Code

Items

CCOA ID Desc.Common COA ID Desc

Mortgages1 - Instrument Type Code.Instrument Type Code Description

Mortgages1 - Product Type Code.Product Type Code Description

Calculations

Item Name	Formula	Description
Current OAS Percent	Current OAS/100	Returns rate in percentage
Current Static Spread Percent	Current Static Spread/100	Returns rate in percentage
Historic OAS Percent	Historic OAS/100	Returns rate in percentage
Historic Statis Spread Percent	Historic Static Spread/100	Returns rate in percentage

Filters

Common COA ID Desc = :Common COA

Product Type Code Description = :Product

Joins

CCOA ID Desc > Mortgages1

Mortgages1 - Product Type Code > Mortgages1

Mortgages1 - Instrument Type Code > Mortgages1

User Prompts

Prompt	User Action
Common COA	Enter the Common COA ID Description on which to report.
Product	Enter the product Type Code Description on which to report.

Part III

Standard Reports for Oracle Reports

The Standard Reports for Oracle Reports are based on SQL scripts. You can use or customize these reports immediately after installation.

The material contained in this book part makes the following assumptions:

- Oracle Reports 6.0 or above is installed

- The .rdf files are installed on the PC of each user running the reports

- You have transformed your data using the Transformation ID

For information on Oracle Reports, refer to the *Oracle Reports Reference Manual or the Oracle Reports Building Reports Manual*. You can download these manuals from http://technet.us.oracle.com. For information on using the Transformation ID, see the "Transformation ID" chapter in the *Oracle Financial Data Manager Balance & Control Reference Guide.*

This part provides a description and the SQL script for each report. The reports are grouped into the following chapters:

17

Overview of the Standard Reports for Oracle Reports

The Oracle Financial Services applications use Oracle Reports to provide advanced reporting. To simplify the use of Oracle Reports, a set of Standard Reports that runs against the Oracle Financial Data Manager (FDM) Reporting Data Mart is available with the installation of FDM.

This chapter provides an overview of the Oracle Reports product and the FDM Standard Reports available for this product. It also discusses modifying the reports to meet the needs of your organization. This information is presented in the following topics:

- Overview of Oracle Reports
- Overview of the Standard Reports
- Modifying the Standard Reports

17.1 Overview of Oracle Reports

Oracle Reports is a powerful enterprise reporting tool to produce production reports and sophisticated Web-based reports. These reports dynamically retrieve, format and distribute database information. Oracle Reports provides many features, including:

- Unlimited data formatting
- Distributes high quality reports via the Internet
- Scalable application server based reporting
- Integrated with the tools of Oracle Business Intelligence

Oracle Reports Application Server based architecture means report consumers require only a Web browser to view reports in industry standard formats. Oracle Reports supports on-demand delivery of high quality reports over the Web through native generation of HTML with cascading style sheets and Adobe Portable Document Format (PDF).

Oracle Reports offers report developers a powerful declarative development environment allowing sophisticated, unbounded reports to be created from scratch in a matter of minutes. The time taken to deliver the complex reporting that enables you to really meet the demands of business is no longer an issue.

17.2 Overview of the Standard Reports

Upon installation, the Standard Reports are available for immediate use. Each of the provided sample reports is a self-contained file. Use the OFSA Universal Demo Database to sample these reports. The Standard Reports are:

Common Reports

Hierarchy Report for Organizational Unit (hierarc. rdf).

Performance Analyzer Reports

Balance Sheet (balance.rdf)

Balance Sheet with Multicurrency (balance_mc.rdf)

Balance Sheet and Income Statement (incbal.rdf)

Balance Sheet and Income Statement with Multicurrency (incbal_mc.rdf)

Income Statement (income.rdf)

Income Statement Multicurrency (income_mc.rdf)

Stratification Report (strat1.rdf)

Rate Manager Reports

Exchange Rate Report (exchng.rdf)

Risk Manager Reports

Detail Cashflow Runoff Report (detcfrun.rdf)

Gap Summary Report (rmgapsum.rdf)

Market Value Report (rmmktval.rdf)

Detail/Summary Cashflow Report (rmcashfl.rdf)

Gap Detail Report (rmgapdet.rdf)

Scenario Income Statement (rmincsce.rdf)

Summary Income Statement (rmincsum.rdf)

Transfer Pricing Reports

Transfer Pricing Income Statement (rminctp.rdf)

17.2.1 Oracle Reports Templates

A report template is different from a report definition. The Template editor allows you to modify existing objects in the template's header, trailer, margin, and body regions. You can customize other reports objects: parameters, report-level triggers, program units, and attached PL/SQL libraries.

You should modify template objects in the Object Navigator, instead of the Template editor. Use the Template editor as a reference, to view the changes you have made to the template objects.

Make general modifications to the default section. The modifications affect all report styles. However, you can override attributes for one or more styles, to provide a flexible inheritance model.

For more information, see Chapter 18, "Predefined Templates".

17.2.2 Exporting From Discoverer

You can export a report from Discoverer to Oracle Reports using the provided templates. The following directions explain how to use a template with Discoverer.

1. Run a Discoverer Workbook.

2. Select File Export from the menu.

3. Select the Oracle Report Definition export format.

4. Click Next after changing the name of the output file.

5. Choose select a custom template file.

6. Enter the location and name of the template file to be used.

7. Click Finish.

Discoverer opens either the Reports Builder or the runtime, and generates the report based on the format definition stored in the selected .rft file.

17.3 Modifying the Standard Reports

The report examples included within the OFSA Universal Demo database provides you with a starting point to design and create reports focused on the needs of your organization.

The following steps provide guidelines for modifying these reports:

1. Determine the purpose of the report and exactly what the information to be presented must contain.

2. Locate the report file that most closely represents this purpose.

3. Save the file to a working directory with a user-defined file name.

   This step ensures the example file is not overwritten.

4. Open Oracle Reports Builder.

5. Open the file from step 2.

6. Modify the report definition and its associated template to meet your business needs.

For detailed information on each report, refer to the appropriate chapter based on report classifications.

18

Predefined Templates

Oracle Reports uses template files to format the output of reports. You specify the template file when you export the report from Oracle Discoverer or you create the report using the Oracle Reports wizard.

This section discusses the predefined template files used in the Standard Reports.

18.1 Template and Guidelines

The FDM-related predefined report templates for Oracle Reports contain most of the necessary formatting information. Standards for Parameter forms cannot be defined within the template as they follow the description of the template files.

The available templates are:

- ofsa_por.tdf - Portrait
- ofsa_ls.tdf - Landscape

The following are illustrations of the templates.

Top of Page (Fonts, Placement)

Illustration	Fonts Used
	Arial Bold 8 (Title Line 1)
	Arial Standard 8 (Title Line 2)
	Arial Standard 8 (Title Line 3)
	Arial Bold 8 (Parameter Field 1)
	Arial Bold 8 (Parameter Field N)

Grid Color and Outline Row and Column Descriptions

Convention	Description
Arial Standard 8	Column Descriptions
Arial Standard 8	Row Descriptions
Arial Bold 8	Summaries
Outline Exact color match	Enables a Web browser to print with outlines when printing row and column descriptions

Cell Values

Cell Value	Description
Values	Arial Standard 8 White background Outline Second Row of gray third box
Summaries	Arial Bold 8 White background Gray Outline second row of gray, third box
Cell outline	A darker color than Row and Column description provides contrast

Cell Formats

Cell	Format	Comment
Dates	MM/DD/RRRR (Four-digit year)	This is a custom date format built using the menu selection Tools > Preferences > Edit masks selection.
Numeric	($NNN,NN0)	This is a custom format produced with the following PL/SQL code within the format trigger PL/SQL code selection: ■ Dollar Format Mask: srw.set_format_mask('$NNN,NN0'); ■ Percentage/Rate Format Mask: srw.set_format_mask('N0.000%');
Percentage/ Rates	(N0.000%)	(Same as for the Numeric cell.)

Bottom of Page #

Date, Format MM/DD/RRRR   Page# of Total Pages   File Name

Time, Format HH:MM:SS AM

HTML Report Escape Properties

Value	Property
Before Report Value	\<html\>
	\<body bgcolor="#ffffff"\>
After Report Value	\</body\>
	\</html\>
Before Page Value	—
After Page Value	\<a href="http://Place your home server name ere"\>Home\</a\>\<hr\>
Before Form Value	\<html\>
	\<body bgcolor="#ffffff"\>
	\<form method=post action="_action_"\>
	\<input name="hidden_run_parameters" type=hidden value="_hidden_"\>
	\<center\>

Parameter Form Standards Used in Developing Reports

Description of report

Arial Bold 8 (Title matches report)

Arial Standard 8 (Instruction Line)

Parameter Value Selection List

Arial Standard 8

(Parameter Description)

18.2 Building the Parameter Form

To build the parameter form, follow these steps:

1. Select Parameter Form Builder.

2. Create desired parameters. Specify whether the parameter will represent a list of values or a static value.

3. Add bind parameters, such as :ParameterName, to the report SQL statements.

4. Add dropdown list boxes, validation codes, format masks, and initial values.

See the Report Builder Help Topics for more information about Parameter forms.

18.3 Customizing the Template File

You can customize the template files to fit your organization. You can modify the existing template files, use one of the template files delivered with the Oracle Reports software, or create your own.

Here are some suggestions for customizing template files delivered with the Standard Reports for Oracle Reports:

1. Replace the existing bitmap image with your company logo.

2. Select View, Layout Section, Margin View from the menu bar.

3. Select File, Import, Image from the menu bar.

4. Import a bitmap of your company logo or insert an ole object into the margin.

5. Adjust the color scheme and fonts to your preferences.

You can add your customized template file to the Reports Builder wizard list of predefined templates. For more information, see "Adding a Template to the Predefined Templates List" in the Report Builder Help.

18.4 Changing the Company Logo

Do the following to change the Oracle bitmap:

1. Doubleclick on Layout model within the Object Navigator Locate Reports Layout Model. to invoke the report editor.

2. Delete the current image using the Report Editor.

3. Select File > Import Image, and then browse to locate the image file you want imported into the Layout Model.

4. Select the file and click OK.

5. Select the Report Editor – parameter form and delete the image.

6. Import the desired image.

19

Oracle FDM Common Reports

The Oracle Financial Data Manager (FDM) Common Reports work on the common components of the Reporting Data Mart. Currently, one report is available in this classification: The Hierarchy Report for Organizational Unit. The Hierarchy Report shows the organizational hierarchy for the OFSA demo database.

Report

Hierarchy Report for Organizational Unit

File

hierarc.rdf

SQL Documentation

The SELECT statement queries Org_Unit-level data at multiple levels. The report presents that data in tabular form.

```
SELECT DISTINCT
    OFS_RPT_HIER_ORG_UNIT.LEAF_DESCRIPTION AS E135481,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_01_DESC AS E135483,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_02_DESC AS E135485,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_03_DESC AS E135487,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_04_DESC AS E135489,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_05_DESC AS E135491
FROM OFS_RPT_HIER_ORG_UNIT OFS_RPT_HIER_ORG_UNIT
ORDER BY OFS_RPT_HIER_ORG_UNIT.LEVEL_05_DESC ASC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_04_DESC ASC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_03_DESC ASC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_02_DESC ASC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_01_DESC ASC,
    OFS_RPT_HIER_ORG_UNIT.LEAF_DESCRIPTION ASC
```

20

Oracle Performance Analyzer Reports

The Oracle Performance Analyzer reports deliver a comprehensive profitability picture across a financial organization based on allocation activities. Additionally, the reports support multicurrency reporting, enabling multinational corporations to analyze financial performance in different currencies.

This chapter provides detailed information on the following reports:

- Balance Sheet
- Balance Sheet with Multicurrency
- Balance Sheet and Income Statement
- Balance Sheet and Income Statement with Multicurrency
- Income Statement
- Income Statement Multicurrency
- Stratification Report

20.1 Balance Sheet

The Balance Sheet report tells the user exactly where the asset/liability/equity distribution of the organization stands at any given time. This report has been modified to include organization unit. It can be further modified to include such items as branches or cost centers, enabling the analyst to concentrate on asset and liability distribution at any enterprise level.

- File name = balance.rdf

SQL Documentation

The SELECT statement pulls and sums data from the account level with respect to the Assets, Liabilities, and Equity classification. The data is then divided by 1000 to simplify the reported information.

```
SELECT OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC AS E101734,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC AS E101736, OFS_RPT_LEDGER_STAT.END_
    DATE AS E101755,SUM(DECODE(OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC,'Assets',
    OFS_RPT_LEDGER_STAT.END_BAL/1000,'Liabilities and Equity',OFS_RPT_LEDGER_
    STAT.END_BAL/1000,'0')) AS E_44
FROM OFS_RPT_HIER_GL_ACCT_IDOFS_RPT_HIER_GL_ACCT_ID, OFS_RPT_LEDGER_STAT
    OFS_RPT_LEDGER_STAT
```

The WHERE statement qualifies and assigns the data as 'Liabilities and Equity' or 'Assets'. The Beginning_Date, Ending_Date, and Org_Unit have all been declared as bind variables for purposes of selecting these values from a drop-down parameter list.

```
WHERE ( ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) = OFS_RPT_LEDGER_STAT.GL_
ACCOUNT_ID ) )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC IN ('Liabilities and
    Equity','Assets') )
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_Date and :Ending_Date
    AND ( OFS_RPT_LEDGER_STAT.ORG_UNIT_ID IN ( SELECT ORG_UNIT_ID
FROM OFS_RPT_HIER_ORG_UNIT
WHERE LEVEL_04_DESC = :Org_Unit ) )
GROUP BY OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC, OFS_RPT_HIER_GL_ACCT_ID.LEVEL_
06_DESC, OFS_RPT_LEDGER_STAT.END_DATE
```

20.2 Balance Sheet with Multicurrency

The Balance Sheet report tells the user exactly where the asset/liability/equity distribution of the organization stands at any given time. This report has been modified to include organization unit. It can be further modified to include such items as branches or cost centers, enabling the analyst to concentrate on asset and liability distribution at any enterprise level.

- ■ File name = balance_mc.rdf

SQL Documentation

The SELECT statement includes a predefined package (OFSA_RATES) that calls a function CONVERT_BALANCE. The CONVERT_BALANCE function takes input balance (End balance of Assets, Liabilities and Equities), From Currency, To

Currency and Effective Date. The function converts the balance based on the exchange rate on the effective date. The Beginning_Date, Org_Unit and the desired Currency have all been declared as bind variables for purposes of selecting these values from a drop-down parameter list.

```
SELECT OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC AS E101734,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC AS E101736,
    OFS_RPT_LEDGER_STAT_MC.END_DATE AS E101755,
    OFSA_RATES.CONVERT_BALANCE(SUM(DECODE(OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_
    DESC,'Assets',OFS_RPT_LEDGER_STAT_MC.END_BAL/1000,'Liabilities and
    Equity',OFS_RPT_LEDGER_STAT_MC.END_BAL/1000,'0')),
    OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD,:to_cur,
    OFS_RPT_LEDGER_STAT_MC.END_DATE) AS E_44
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID, OFS_RPT_HIER_ORG_UNIT OFS_
RPT_HIER_ORG_UNIT, OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ((OFS_RPT_HIER_ORG_UNIT.ORG_UNIT_ID = OFS_RPT_LEDGER_STAT_MC.ORG_UNIT_ID)
    AND (OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID = OFS_RPT_LEDGER_STAT_MC.GL_
    ACCOUNT_ID ) )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC IN ('Liabilities and
    Equity','Assets') ) AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE = :Ending_Date )
    AND OFS_RPT_HIER_ORG_UNIT.ORG_UNIT_ID IN (
    SELECT ORG_UNIT_ID
    FROM OFS_RPT_HIER_ORG_UNIT
    WHERE LEVEL_04_DESC = :Org_Unit)
    GROUP BY OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC, OFS_RPT_HIER_GL_ACCT_
    ID.LEVEL_05_DESC, OFS_RPT_HIER_ORG_UNIT.LEVEL_05_DESC, OFS_RPT_LEDGER_STAT_
    MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
```

20.3 Balance Sheet and Income Statement

The Balance Sheet and Income Statement enables the user to see the institutional high level finances on one page. All the information from the balance sheet and income statement is combined to instantly show the current position of the bank. Also, there is a variety of ratios included to help the analyst with instant decision making. This report provides great opportunity for you to add additional information that is pertinent to your organization.

■ File name = incbal.rdf

SQL Documentation

This query was copied and pasted from a Discoverer workbook, then modified for reporting. The data has been pivoted, and tokens have been added for formatting

and sorting. The format of the data within the source table is not compatible with a matrix format for reporting. Therefore, the data has been pivoted by a series of SELECT statements unioned together.

```
-- Query 01 --
SELECT 1, "GroupFlag", 10 "Sort", ' ' "FormatString",
    OFS_RPT_LEDGER_STAT.END_DATE "End Date", 'BALANCE SHEET ($000,000s)' "Row_
    Description", To_Number(Null) "Value"
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 02 --
SELECT 1 , 20 , ' ' , OFS_RPT_LEDGER_STAT.END_DATE "End Date",
      'ASSETS:', To_Number(Null)
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 03 --
SELECT 0 , 30 , '$' , OFS_RPT_LEDGER_STAT.END_DATE "End Date",
  'Total Earning Assets' , SUM ( OFS_RPT_LEDGER_STAT.END_BAL / 1000000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_ DESC = 'Earning Assets')))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND (OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date )
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 04 --
SELECT 0 , 40 , '$' , OFS_RPT_LEDGER_STAT.END_DATE "End Date",
  'Non-Earning Assets' , SUM (OFS_RPT_LEDGER_STAT.END_BAL / 1000000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
    FROM   OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_ DESC = 'Non-
                   Earning Assets')))
```

```
        AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 05 --
SELECT 1, 50 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Assets' ,
       SUM(OFS_RPT_LEDGER_STAT.END_BAL/1000000)
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_ DESC = 'Assets')))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY FS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 06 --
SELECT 1 , 60 , ' ' , OFS_RPT_LEDGER_STAT.END_DATE "End Date",
  'LIABILITIES:', To_Number(Null)
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 07 --
SELECT 0, 70 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Demand Deposits',
  SUM ( OFS_RPT_LEDGER_STAT.END_BAL / 1000000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_ DESC = 'Total
                Demand Deposits')))
    AND OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date )
GROUP BY FS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 08 --
SELECT 0, 80 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Interest Bearing
Funds', SUM( OFS_RPT_LEDGER_STAT.END_BAL / 1000000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
```

```
            WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
                WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_ DESC = 'Total
                            Interest Bearing Funds')))
                AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and
                    :Ending_Date )
            GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
            UNION

            -- Query 09 --
            SELECT 0, 90 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Deposits' , SUM( OFS_
            RPT_LEDGER_STAT.END_BAL / 1000000 )
            FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
            WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                FROM   OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
                WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_ DESC
                            IN ( 'Total Demand Deposits',
                                'Total Interest Bearing Funds'  ))))
                AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
            GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
            UNION
            -- Query 010 --
            SELECT 0, 100 , '$' , OFS_RPT_LEDGER_STAT.END_DATE ,
                'Total Other Liabilities', SUM( OFS_RPT_LEDGER_STAT.END_BAL / 1000000 )
            FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
            WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                        FROM   OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
                        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_ DESC
                            IN ('Acceptances Outstanding', 'Other Liabilities' ))))
                AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
            GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
            UNION

            -- Query 11 --
            SELECT 1, 110 , '$' , OFS_RPT_LEDGER_STAT.END_DATE ,
                'Total Liabilities' , SUM( OFS_RPT_LEDGER_STAT.END_BAL / 1000000)
            FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
            WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
```

```
            FROM   OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
            WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_ DESC = 'Total
                   Liabilities' )))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 12 --
SELECT 1, 120, ' ', OFS_RPT_LEDGER_STAT.END_DATE, 'INCOME STATEMENT ($000s)',
To_Number(Null)
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 13 --
SELECT 1, 130, ' ', OFS_RPT_LEDGER_STAT.END_DATE, 'INCOME:' ,
      To_Number(Null)
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 14 --
SELECT 0, 140 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Interest Income' ,
      SUM( OFS_RPT_LEDGER_STAT.INTEREST / 1000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC =
                  'Interest Income' )))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 15 --
SELECT 0, 150 , '%' , OFS_RPT_LEDGER_STAT.END_DATE ,
      'Yield on Earning Assets', SUM( OFS_RPT_LEDGER_STAT.INTEREST ) /
      SUM(OFS_RPT_LEDGER_STAT.END_BAL ) * 12 * 100
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
```

```
                WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
                    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC = 'Interest Expense' )
                    OR (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC = 'Earning Assets' )))
                    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
                GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
                UNION

                -- Query 16 --
                SELECT 0, 160, '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Interest Expense',
                      SUM( - OFS_RPT_LEDGER_STAT.INTEREST / 1000 )
                FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
                WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
                    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC = 'Interest Expense' )))
                    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
                GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
                UNION

                -- Query 17 --
                SELECT 0, 170 , '%' , OFS_RPT_LEDGER_STAT.END_DATE ,
                    'Interest Expense Rate' , SUM( - OFS_RPT_LEDGER_STAT.INTEREST ) /
                    SUM( OFS_RPT_LEDGER_STAT.END_BAL ) * 12 * 100
                FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
                WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
                    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
                    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
                    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC = 'Interest Expense' )
                    OR (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC IN
                    ( 'Total Demand Deposits' , 'Total Interest Bearing Funds' ))))
                    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
                    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
                GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
                UNION

                -- Query 18 --
                SELECT 1, 180 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Net Margin' ,
                      SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC ,
                         'Interest Income', OFS_RPT_LEDGER_STAT.INTEREST/1000, 0 ))
                       - SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC ,
                         'Interest Expense', OFS_RPT_LEDGER_STAT.INTEREST/1000, 0 ))
```

```
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID,
      OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) =
          OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID ) )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC IN ( 'Interest
        Income', 'Interest Expense' ) )
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION
-- Query 19 --
SELECT 0, 190, '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Non-Interest Income',
SUM( OFS_RPT_LEDGER_STAT.NON_INT_INC / 1000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
            SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
            FROM  OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
            WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC = 'Total
                   Non-Interest Income' )))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 20 --
SELECT 0, 200 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Total Non-interest
Expense', SUM( - OFS_RPT_LEDGER_STAT.NON_INT_EXP / 1000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
            SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
            FROM  OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
            WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC = 'Total
                   Non-interest Expense' )))
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Begining_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 21 --
SELECT 1, 210 , '$' , OFS_RPT_LEDGER_STAT.END_DATE ,
      'Net Non-Interest Income', SUM( OFS_RPT_LEDGER_STAT.NON_INT_INC / 1000 ) -
      SUM( OFS_RPT_LEDGER_STAT.NON_INT_EXP / 1000 )
FROM OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID IN (
            SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ ID
```

```
                   FROM   OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_ GL_ACCT_ID
               WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC
                  IN ( 'Total Non-Interest Income' , 'Total Non-
                       interest Expense' ))))
     AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
     AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Beging_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
UNION

-- Query 21 --
SELECT 1, 220 , '$' , OFS_RPT_LEDGER_STAT.END_DATE , 'Net Income',
     SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC , 'Total Non-Interest
     Income', OFS_RPT_LEDGER_STAT.NON_INT_INC / 1000 , 0 ) )
       - SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC , 'Total Non-interest
     Expense', OFS_RPT_LEDGER_STAT.NON_INT_EXP/1000 , 0 ) )
       + SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC , 'Interest Income',
     OFS_RPT_LEDGER_STAT.INTEREST/1000 , 0 ) )
       - SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_ DESC,
                     'Interest Expense', OFS_RPT_LEDGER_STAT.INTEREST / 1000, 0))
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID,
     OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
WHERE ( ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) =
          OFS_RPT_LEDGER_STAT.GL_ACCOUNT_ID ) )
     AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC
       IN ( 'Total Non-Interest Income', 'Total Non-interest Expense',
       'Interest Income', 'Interest Expense' ) )
     AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
     AND ( OFS_RPT_LEDGER_STAT.END_DATE BETWEEN :Beging_ Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT.END_DATE
ORDER BY 4, 2
```

20.4 Balance Sheet and Income Statement with Multicurrency

The Balance Sheet and Income Statement with Multicurrency report enables the user to see the institutional high level finances on one page. All the information from the balance sheet and income statement is combined to instantly show the current position of the bank. Also, there is a variety of ratios included to help the analyst with instant decision making. This report provides great opportunity for the user to add additional information that is pertinent to the institution.

- File name = incbal_mc.rdf

SQL Documentation

This query was copied and pasted from a Discoverer workbook, then modified for reporting. The data has been pivoted, and tokens have been added for formatting and sorting. The format of the data within the source table is not compatible with a matrix format for reporting. Therefore, the data has been pivoted by a series of SELECT statements unioned together.

```
-- Query 01 --
SELECT 1, "GroupFlag", 10, "Sort", ' ', "FormatString", OFS_RPT_LEDGER_STAT_
MC.END_DATE "End Date", 'BALANCE SHEET (000,000s)' "Row_Description", To_
Number(Null) "Value"
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND (OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date)
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
UNION

-- Query 02 --
SELECT 1, 20, ' ', OFS_RPT_LEDGER_STAT_MC.END_DATE "End Date", 'ASSETS:', To_
Number(Null)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 ) AND (OFS_RPT_LEDGER_
STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_Date)
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
UNION

-- Query 03 --
SELECT 0, 30, '$',OFS_RPT_LEDGER_STAT_MC.END_DATE "End Date",
    'Total Earning Assets', OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_
    MC.END_BAL / 1000000 ), ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
            SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
            FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
            WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC = 'Earning Assets')))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 04 --
SELECT 0, 40, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE "End Date",
```

```
        'Non-Earning Assets', OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_
        MC.END_BAL / 1000000 ), ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
        FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC = 'Non-Earning Assets')))
        AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
        Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE , OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_
CD
UNION

-- Query 05 --
SELECT 1, 50, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Total Assets',
        OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
        ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
        FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_DESC = 'Assets')))
        AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND (OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
        Date)
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 06 --
SELECT 1 , 60 , ' ' , OFS_RPT_LEDGER_STAT_MC.END_DATE "End Date",
'LIABILITIES:', To_Number(Null)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE (OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
        Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
UNION

-- Query 07 --
SELECT 0, 70 , '$' , OFS_RPT_LEDGER_STAT_MC.END_DATE , 'Total Demand Deposits',
        OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
        ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE (OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
```

```
        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
        FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC = 'Total Demand Deposits')))
        AND (OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND
        (OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 08 --
SELECT 0, 80, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE,
        'Total Interest Bearing Funds', OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_
        LEDGER_STAT_MC.END_BAL / 1000000 ), ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
        FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC = 'Total Interest Bearing
        Funds')))
        AND (OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
        Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 09 --
SELECT 0, 90, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Total Deposits',
        OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
        ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE (OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
        SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
        FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
        WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC
        IN ('Total Demand Deposits', 'Total Interest Bearing Funds' ))))
        AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
        Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 010 --
SELECT 0, 100 '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Total Other Liabilities',
        OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
        ISO_CURRENCY_CD, :TO_CUR, END_DATE)
```

```
               FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
               WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
                   SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
                   FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
                   WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC
                   IN ( 'Acceptances Outstanding', 'Other Liabilities' ))))
                   AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
                   AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
                   Date )
               GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
                   UNION


               -- Query 11 --
               SELECT 1, 110, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Total Liabilities',
                   OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
                   ISO_CURRENCY_CD, :TO_CUR, END_DATE)
               FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
               WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
                   SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
                   FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
                   WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC = 'Total Liabilities' )))
                   AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
                   AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
                   Date )
               GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
               UNION

               -- Query 12 --
               SELECT 1, 120, ' ', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'INCOME STATEMENT (000s)',
                   To_Number(Null)
               FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
               WHERE ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
               AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_Date)
               GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
               UNION

               -- Query 13 --
               SELECT 1, 130, ' ', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'INCOME:',
                   To_Number(Null)
               FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
               WHERE ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
               AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_Date )
               GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
               UNION
```

```
-- Query 14 --
SELECT 0, 140 , '$' , OFS_RPT_LEDGER_STAT_MC.END_DATE ,'Interest Income',
    OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL / 1000000 ),
    ISO_CURRENCY_CD,:TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Interest Income' )))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE,OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 15 --
SELECT 0, 150 , '%', OFS_RPT_LEDGER_STAT_MC.END_DATE,
    'Yield on Earning Assets', SUM( OFS_RPT_LEDGER_STAT_MC.INTEREST )
    / SUM( OFS_RPT_LEDGER_STAT_MC.END_BAL ) * 12 * 100
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Interest Expense' )
    OR (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_05_DESC = 'Earning Assets' )))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
UNION

-- Query 16 --
SELECT 0, 160, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Interest Expense',
    OFSA_RATES.CONVERT_BALANCE(SUM( - OFS_RPT_LEDGER_STAT_MC.INTEREST / 1000 ),
    ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Interest Expense' )))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION
```

```
-- Query 17 --
SELECT 0, 170, '%' , OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Interest Expense Rate',
    SUM( - OFS_RPT_LEDGER_STAT_MC.INTEREST ) / SUM( OFS_RPT_LEDGER_STAT_MC.END_
    BAL ) * 12 * 100
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Interest Expense' )
    OR (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC IN
    ( 'Total Demand Deposits' , 'Total Interest Bearing Funds' ))))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE
UNION

-- Query 18 --
SELECT 1, 180, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Net Margin',
    OFSA_RATES.CONVERT_BALANCE(SUM(DECODE(OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC,
    'Interest Income', OFS_RPT_LEDGER_STAT_MC.INTEREST/1000, 0 ))
    - SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC, 'Interest Expense',
    OFS_RPT_LEDGER_STAT_MC.INTEREST/1000, 0 )), ISO_CURRENCY_CD, :TO_CUR, END_
    DATE)
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID, OFS_RPT_LEDGER_STAT_MC
OFS_RPT_LEDGER_STAT_MC
WHERE ( ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) = OFS_RPT_LEDGER_STAT_MC.GL_
ACCOUNT_ID ) )
AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC IN ( 'Interest Income' , 'Interest
Expense' ) )
AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 19 --
SELECT 0, 190, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE,
    'Total Non-Interest Income', OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_
    STAT_MC.NON_INT_INC / 1000 ), ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Total Non-Interest Income'
```

```
    )))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION
-- Query 20 --
SELECT 0, 200, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE,
    'Total Non-interest Expense',OFSA_RATES.CONVERT_BALANCE(SUM( - OFS_RPT_
    LEDGER_STAT_MC.NON_INT_EXP / 1000 ), ISO_CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC = 'Total Non-interest Expense'
    )))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 21 --
SELECT 1, 210, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Net Non-Interest Income',
    OFSA_RATES.CONVERT_BALANCE(SUM( OFS_RPT_LEDGER_STAT_MC.NON_INT_INC / 1000 )
    - SUM( OFS_RPT_LEDGER_STAT_MC.NON_INT_EXP / 1000 ), ISO_CURRENCY_CD, :TO_
    CUR, END_DATE)
FROM OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( OFS_RPT_LEDGER_STAT_MC.GL_ACCOUNT_ID IN (
    SELECT OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID
    FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID
    WHERE (OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC IN
    ( 'Total Non-Interest Income' , 'Total Non-interest Expense' ))))
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
UNION

-- Query 22 --
SELECT 1, 220, '$', OFS_RPT_LEDGER_STAT_MC.END_DATE, 'Net Income',
    OFSA_RATES.CONVERT_BALANCE(SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_
    DESC, 'Total Non-Interest Income', OFS_RPT_LEDGER_STAT_MC.NON_INT_INC /
    1000, 0))
     - SUM(DECODE(OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC,
```

```
                'Total Non-interest Expense', OFS_RPT_LEDGER_STAT_MC.NON_INT_EXP/1000,
                0))
        + SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC ,
            'Interest Income', OFS_RPT_LEDGER_STAT_MC.INTEREST/1000 , 0 ) )
        - SUM(DECODE( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC,
            'Interest Expense', OFS_RPT_LEDGER_STAT_MC.INTEREST / 1000 , 0 )), ISO_
            CURRENCY_CD, :TO_CUR, END_DATE)
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID,
         OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ( ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) = OFS_RPT_LEDGER_STAT_MC.GL_
ACCOUNT_ID ) )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC IN
        ( 'Total Non-Interest Income', 'Total Non-interest Expense', 'Interest
        Income', 'Interest Expense' ) )
    AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_LEDGER_STAT_MC.END_DATE BETWEEN :Begining_Date and :Ending_
    Date )
GROUP BY OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
ORDER BY 4, 2
```

20.5 Income Statement

The Income Statement shows profitability at any given moment. This report has
been modified to include organizational units. Similar to the Balance Sheet, this
report can be modified to include subdivisions or cost centers.

- File name = income.rdf

SQL Documentation

The statement is similar to the Balance Sheet report in that the data is pulled from
the declared account levels within the Ledger_Stat table. The specified data is then
divided by 1000 to simplify the reported information.

```
SELECT OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC AS E101730,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC AS E101732,
    OFS_RPT_LEDGER_STAT.END_DATE AS E101755, SUM(( DECODE('Level 4
    Description','Interest Expense',-1,'Total Non-Interest Expense',-1,'Loan
    Loss Provision',-1,1) ) * (OFS_RPT_LEDGER_STAT.INTEREST +
    OFS_RPT_LEDGER_STAT.NON_INT_EXP+OFS_RPT_LEDGER_STAT.NON_INT_INC )/1000) AS
    E_14
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID,
    OFS_RPT_HIER_ORG_UNIT OFS_RPT_HIER_ORG_UNIT,
    OFS_RPT_LEDGER_STAT OFS_RPT_LEDGER_STAT
```

The WHERE statement qualifies and assigns the data as 'Net Margin',' Non-Interest Income' or 'Security Gain/Loss'. The 'Beginning_Date', 'End_Date', and 'Org_Unit' have all been declared as bind variables for purposes of selecting these values from a drop-down parameter list.

```
WHERE (( OFS_RPT_HIER_ORG_UNIT.ORG_UNIT_ID(+) = OFS_RPT_LEDGER_STAT.ORG_UNIT_ID)
    AND ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) = OFS_RPT_LEDGER_STAT.GL_
    ACCOUNT_ID ) )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC IN ('Net Margin','Net
    Non-Interest Income','Security Gain/Loss') )
    AND ( OFS_RPT_LEDGER_STAT.CONSOLIDATION_CD = 100 )
    AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_VALUE(+) <> '-99000L01' )
    AND ( ofs_rpt_hier_org_unit.level_04_desc =:level_04_desc )
    AND ( ofs_rpt_ledger_stat.end_date between :Begining_Date and :Ending_Date )
GROUP BY OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_04_DESC,
    OFS_RPT_HIER_ORG_UNIT.LEVEL_05_DESC,
    OFS_RPT_LEDGER_STAT.END_DATE
```

20.6 Income Statement Multicurrency

The Common Income statement shows profitability at any given moment. This report has been modified to include organizational units. Similar to the Balance Sheet, this report can be modified to include subdivisions or cost centers.

- File name = income_mc.rdf

SQL Documentation

The SELECT statement includes a predefined package (OFSA_RATES) that calls a function CONVERT_BALANCE.   The CONVERT_BALANCE function takes input balance (End balance of Assets, Liabilities and Equities), From Currency, To Currency and Effective Date. The function converts the balance based on the exchange rate on the effective date. The Start Date, End Date, Org_Unit and the desired Currency have all been declared as bind variables for purposes of selecting these values from a drop-down parameter list.

```
SELECT OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC AS E101730,
    OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC AS E101732,
    OFS_RPT_LEDGER_STAT_MC.END_DATE AS E101755,
    OFSA_RATES.CONVERT_BALANCE(SUM(( DECODE('Level 4 Description','Interest
    Expense',-1,'Total Non-Interest Expense',-1,'Loan Loss Provision',-1,1) )* (
```

```
        OFS_RPT_LEDGER_STAT_MC.INTEREST+ OFS_RPT_LEDGER_STAT_MC.NON_INT_EXP+OFS_RPT_
        LEDGER_STAT_MC.NON_INT_INC )/1000),
        OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD,:to_cur,END_DATE)  AS E_14
FROM OFS_RPT_HIER_GL_ACCT_ID OFS_RPT_HIER_GL_ACCT_ID,
        OFS_RPT_HIER_ORG_UNIT OFS_RPT_HIER_ORG_UNIT,
        OFS_RPT_LEDGER_STAT_MC OFS_RPT_LEDGER_STAT_MC
WHERE ((OFS_RPT_HIER_ORG_UNIT.ORG_UNIT_ID(+) = OFS_RPT_LEDGER_STAT_MC.ORG_UNIT_
ID)
        AND ( OFS_RPT_HIER_GL_ACCT_ID.GL_ACCOUNT_ID(+) = OFS_RPT_LEDGER_STAT_MC.GL_
        ACCOUNT_ID ) )
        AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC IN ('Net Margin','Net
        Non-Interest Income','Security Gain/Loss') )
        AND ( OFS_RPT_LEDGER_STAT_MC.CONSOLIDATION_CD = 100 )
        AND ( OFS_RPT_HIER_GL_ACCT_ID.LEVEL_06_VALUE(+) <> '-99000L01' )
        AND
        ( ofs_rpt_hier_org_unit.level_04_desc =:level_04_desc )
        AND ( ofs_rpt_ledger_stat_MC.end_date between :Begining_Date and :Ending_
        Date )
GROUP BY OFS_RPT_HIER_GL_ACCT_ID.LEVEL_03_DESC,
        OFS_RPT_HIER_GL_ACCT_ID.LEVEL_04_DESC,
        OFS_RPT_HIER_ORG_UNIT.LEVEL_04_DESC,
        OFS_RPT_HIER_ORG_UNIT.LEVEL_05_DESC,
        OFS_RPT_LEDGER_STAT_MC.END_DATE, OFS_RPT_LEDGER_STAT_MC.ISO_CURRENCY_CD
```

20.7 Stratification Report

Mortgages by rate tier. This report depicts mortgage balances by rate tier. All
Organizational units, all products for all time periods are summed within the
database to reflect a tiered view of mortgage rates. The Report enables the user to
focus on the potential mortgage refinancing impact on the organizational balance
sheet. It can be modified to reflect any stratification necessary.

- File name = stra1.rdf

SQL Documentation

```
SELECT DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier01Low ), :Tier02Low
- .0001 ),MORTGAGES.CUR_GROSS_RATE, 'Mortgage tier 01 (' || To_Char(:Tier01Low)
|| ' - ' || To_Char( :Tier02Low - .0001 ) || ')' ,
    DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier02Low ), :Tier03Low -
    .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 02 (' || To_
    Char(:Tier02Low) || ' - ' || To_Char( :Tier03Low - .0001 ) || ')' ,
    DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier03Low ), :Tier04Low -
    .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 03 (' || To_
    Char(:Tier03Low) || ' - ' || To_Char( :Tier04Low - .0001 ) || ')' ,
```

```
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier04Low ), :Tier05Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 04 (' || To_
        Char(:Tier04Low) || ' - ' || To_Char( :Tier05Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier05Low ), :Tier06Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 05 (' || To_
        Char(:Tier05Low) || ' - ' || To_Char( :Tier06Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier06Low ), :Tier07Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 06 (' || To_
        Char(:Tier06Low) || ' - ' || To_Char( :Tier07Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier07Low ), :Tier08Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 07 (' || To_
        Char(:Tier07Low) || ' - ' || To_Char( :Tier08Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier08Low ), :Tier09Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 08 (' || To_
        Char(:Tier08Low) || ' - ' || To_Char( :Tier09Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier09Low ), :Tier10Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE, 'Mortgage tier 09 (' || To_
        Char(:Tier09Low) || ' - ' || To_Char( :Tier10Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier10Low ), 99 - .0001
        ),MORTGAGES.CUR_GROSS_ RATE, 'Mortgage tier 10 (' || To_ Char(:Tier10Low) ||
        ' - 99.9999)', 'Mortgage tier 11 Other' )))))))))) "Tier Description",
        SUM(MORTGAGES.CUR_NET_BOOK_BAL_C)
FROM MORTGAGES MORTGAGES
GROUP BY DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier01Low ),
:Tier02Low - .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 01 (' || To_
Char(:Tier01Low) || ' - ' || To_Char( :Tier02Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier02Low ), :Tier03Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 02 (' || To_
        Char(:Tier02Low) || ' - ' || To_Char( :Tier03Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier03Low ), :Tier04Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 03 (' || To_
        Char(:Tier03Low) || ' - ' || To_Char( :Tier04Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier04Low ), :Tier05Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 04 (' || To_
        Char(:Tier04Low) || ' - ' || To_Char( :Tier05Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier05Low ), :Tier06Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 05 (' || To_
        Char(:Tier05Low) || ' - ' || To_Char( :Tier06Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier06Low ), :Tier07Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 06 (' || To_
        Char(:Tier06Low) || ' - ' || To_Char( :Tier07Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier07Low ), :Tier08Low -
        .0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 07 (' || To_
        Char(:Tier07Low) || ' - ' || To_Char( :Tier08Low - .0001 ) || ')' ,
        DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier08Low ), :Tier09Low -
```

```
.0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 08 (' || To_
Char(:Tier08Low) || ' - ' || To_Char( :Tier09Low - .0001 ) || ')' ,
DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier09Low ), :Tier10Low -
.0001 ),MORTGAGES.CUR_ GROSS_RATE,  'Mortgage tier 09 (' || To_
Char(:Tier09Low) || ' - ' || To_Char( :Tier10Low - .0001 ) || ')' ,
DECODE(LEAST(GREATEST(MORTGAGES.CUR_GROSS_RATE, :Tier10Low ), 99 - .0001
),MORTGAGES.CUR_GROSS_ RATE,  'Mortgage tier 10 (' || To_ Char(:Tier10Low)
|| ' - 99.9999)' ,'Mortgage tier 11 Other' ))))))))))
```

21

Oracle FDM Rate Manager Reports

The Rate Manager reports provide interest rate and exchange rate information for scenario forecasting exercises. The reports support the multicurrency reporting within FDM, including the current European Monetary Union structure and legislation for the Euro.

Currently, only one report is available for Rate Manager: the Exchange Rate report. This report shows account level gap runoff information for a single start date.

Report Name

Exchange Rate Report

File Name

exchng.rdf

SQL Documentation

```
SELECT OFSA_EXCHANGE_RATE_HIST.FROM_CURRENCY_CD AS E150659,
     OFSA_EXCHANGE_RATE_HIST.TO_CURRENCY_CD AS E150660,
     OFSA_EXCHANGE_RATE_HIST.EFFECTIVE_DATE AS E150661,
     OFSA_EXCHANGE_RATE_HIST.EXCHANGE_RATE AS E150662
FROM OFSA_EXCHANGE_RATE_HIST OFSA_EXCHANGE_RATE_HIST
WHERE OFSA_EXCHANGE_RATE_HIST.EFFECTIVE_DATE = :effective_dt
```

22

Oracle Risk Manager Reports

Using both deterministic and stochastic methods, Oracle Risk Manager produces insightful value-added information for profitability scenario-based processing results and modern financial models to forecast various financial elements. Additionally, the reports support consolidated currency reporting with exchange rate gains and losses. This support enables multinational corporations to manage their exchange rate risk in the volatile global environment.

This chapter provides detailed information on the following reports:

- Detail Cashflow Runoff Report
- Gap Summary Report
- Market Value Report
- Detail/Summary Cashflow Report
- Gap Detail Report
- Scenario Income Statement
- Summary Income Statement

22.1 Detail Cashflow Runoff Report

This report shows the detailed cashflow runoff by account. It is used to report the transactions created by the cashflow engine for the month.

File Name

detcfrun.rdf

SQL Documentation

This query was copied and pasted from a Discoverer workbook, then modified for reporting.

```
SELECT DISTINCT OFSA_PROCESS_CASH_FLOWS.RESULT_SYS_ID AS E165682,
    OFSA_PROCESS_CASH_FLOWS.ID_NUMBER AS E165687, OFSA_PROCESS_CASH_FLOWS.CASH_
    FLOW_DATE AS E165688,RDM_HIER_PRODUCT.LEAF_DESCRIPTION AS E165710,
    DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,60,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_739,
    DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,100,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_735,
    DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,190,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_731,
    DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,250,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_727,
    ( DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,210,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0))+(DECODE(OFSA_PROCESS_CASH_
    FLOWS.FINANCIAL_ELEM_ID,430, OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0)) AS E_
    723, DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,430,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_719,
    DECODE(OFSA_PROCESS_CASH_FLOWS.FINANCIAL_ELEM_ID,210,
    OFSA_PROCESS_CASH_FLOWS.FLOAT_VALUE,0) AS E_715
FROM OFSA_PROCESS_CASH_FLOWS OFSA_PROCESS_CASH_FLOWS, RDM_HIER_PRODUCT RDM_HIER_
PRODUCT
WHERE ((OFSA_PROCESS_CASH_FLOWS.PRODUCT_LEAF_NODE = RDM_HIER_PRODUCT.COMMON_COA_
ID))
    AND ( OFSA_PROCESS_CASH_FLOWS.CASH_FLOW_CD <> 32 )
ORDER BY OFSA_PROCESS_CASH_FLOWS.CASH_FLOW_DATE ASC
```

22.2 Gap Summary Report

The Gap Summary report shows both balance and off-balance sheet gap positions for a single start date.

File Name

rmgapsum.rdf

SQL Documentation

Each SELECT statement is used to declare a level of hierarchy for the report. The first SELECT statement assigns on a 'Total Rollup' level, '0'code for both level 03 and 04 descriptions. Subsequently through each SELECT statement, another level of code will be assigned for the type of data queried.

Ultimately, the SELECT statements will be unioned, and then sorted based on the
assigned hierarchy codes, in order to create the desired row description hierarchy.

```
-- Query 1 --
SELECT 0 Level_03_Sort,0 Level_04_sort,LEVEL_05_DESC Row_Description,
    START_DATE, END_DATE, SUM(GAP_RUNOFF)
FROM OFS_RPT_HIER_COMMON_COA_ID  OFS_RPT_HIER_COMMON_COA_ID ,
    OFS_RPT_RES_DTL_1000018095_G OFS_RPT_RES_DTL_1000018095_G,
    OFSA_RESULT_SCENARIO RESULT_SCENARIO
WHERE RESULT_SCENARIO.SCENARIO_NUM = OFS_RPT_RES_DTL_1000018095_G.SCENARIO_NUM
    AND OFS_RPT_HIER_COMMON_COA_ID .COMMON_COA_ID = OFS_RPT_RES_DTL_1000018095_
    G.TMCOA_ID AND LEVEL_06_DESC IN (('Total Liabilities '|| '&' || '
    Capital'),'Total Assets') AND RESULT_SYS_ID = 1000018095
GROUP BY LEVEL_05_DESC, START_DATE, END_DATE, DESCRIPTION
UNION

-- Query 2 --
SELECT 0 Level_03_Sort,
    Decode(LEVEL_06_DESC, 'Total Assets', 1, 2) Level_04_Sort,
    ' ' || LEVEL_06_DESC, START_DATE, END_DATE, SUM(GAP_RUNOFF)
FROM OFS_RPT_HIER_COMMON_COA_ID  OFS_RPT_HIER_COMMON_COA_ID ,
    OFS_RPT_RES_DTL_1000018095_G OFS_RPT_RES_DTL_1000018095_G,
    OFSA_RESULT_SCENARIO RESULT_SCENARIO
WHERE RESULT_SCENARIO.SCENARIO_NUM = OFS_RPT_RES_DTL_1000018095_G.SCENARIO_NUM
    AND OFS_RPT_HIER_COMMON_COA_ID .COMMON_COA_ID = OFS_RPT_RES_DTL_1000018095_
    G.TMCOA_ID AND OFS_RPT_HIER_COMMON_COA_ID .LEVEL_04_DESC IN (('Total
    Liabilities ' || '&' || ' Capital'),'Total Assets') AND RESULT_SYS_ID =
    1000018095
GROUP BY LEVEL_06_DESC, START_DATE, END_DATE
UNION

-- Query 3 --
SELECT LEVEL_03_SORT,decode (LEVEL_06_DESC, 'Total Assets', 1, 2) Level_04_Sort,
    '   ' || OFS_RPT_HIER_COMMON_COA_ID .LEVEL_03_DESC, START_DATE, END_DATE,
    sum(GAP_RUNOFF) as Gap_Runoff
FROM OFS_RPT_HIER_COMMON_COA_ID  OFS_RPT_HIER_COMMON_COA_ID,
    OFS_RPT_RES_DTL_1000018095_G OFS_RPT_RES_DTL_1000018095_G, OFSA_RESULT_
    SCENARIO RESULT_SCENARIO,
    (SELECT ROWNUM AS LEVEL_03_SORT, LEVEL_03_DESC
    FROM (SELECT DISTINCT LEVEL_03_DESC
        FROM OFS_RPT_HIER_COMMON_COA_ID OFS_RPT_HIER_COMMON_COA_ID,
        OFS_RPT_RES_DTL_1000018095_G OFS_RPT_RES_DTL_1000018095_G,
        OFSA_RESULT_SCENARIO RESULT_SCENARIO
        WHERE RESULT_SCENARIO.SCENARIO_NUM = OFS_RPT_RES_DTL_1000018095_
    G.SCENARIO_NUM
```

```
        AND OFS_RPT_HIER_COMMON_COA_ID.COMMON_COA_ID = OFS_RPT_RES_DTL_
    1000018095_G.TMCOA_ID
        AND LEVEL_06_DESC IN (('Total Liabilities ' || '&' || '
    Capital'),'Total Assets') AND RESULT_SYS_ID = 1000018095
    )) LOOK_UP_KEY
 WHERE RESULT_SCENARIO.SCENARIO_NUM = OFS_RPT_RES_DTL_1000018095_G.SCENARIO_NUM
    AND OFS_RPT_HIER_COMMON_COA_ID .COMMON_COA_ID = OFS_RPT_RES_DTL_1000018095_
    G.TMCOA_ID AND LEVEL_06_DESC IN (('Total Liabilities ' || '&' || '
    Capital'),'Total Assets') AND RESULT_SYS_ID = 1000018095
    AND LOOK_UP_KEY.LEVEL_03_DESC = OFS_RPT_HIER_COMMON_COA_ID.LEVEL_03_DESC
GROUP BY LEVEL_03_SORT,decode(LEVEL_06_DESC, 'Total Assets', 1, 2),
    OFS_RPT_HIER_COMMON_COA_ID.LEVEL_03_DESC, LEVEL_06_DESC, START_DATE, END_
    DATE

-- Sort Order --
Order by Start_Date, Level_04_Sort, Level_03_Sort
```

22.3 Market Value Report

The Market Value Report shows product level information about market values for
a given start date.

- File name = rmmktval.rdf

SQL Documentation

The SELECT statement queries 'Market Value' and 'Duration' product level data.

```
SELECT RESULT_BUCKET.FROM_DATE_001-1 AS E_26,
    RESULT_BUCKET.FROM_DATE_001-1 AS E_23,
    OFS_RPT_HIER_COMMON_COA_ID.LEAF_DESCRIPTION AS E103119, RESULT_
    SCENARIO.DESCRIPTION AS E103772, SUM(RESULT_MASTER.MARKET_VALUE) AS E103502_
    SUM,DECODE(SUM(RESULT_MASTER.MARKET_VALUE),0,0,
    SUM(RESULT_MASTER.DURATION)/SUM(RESULT_MASTER.MARKET_VALUE)) AS E_17
FROM OFS_RPT_HIER_COMMON_COA_ID OFS_RPT_HIER_COMMON_COA_ID,
    OFSA_RESULT_MASTER RESULT_MASTER,OFSA_RESULT_BUCKET RESULT_BUCKET,
    OFSA_RESULT_SCENARIO RESULT_SCENARIO
```

The WHERE statement qualifies the data as well as setting the bind variables. The
Scenario Description and As of-Date have been declared as bind variables for
purposes of selecting these values from a drop-down parameter list.

```
WHERE OFS_RPT_HIER_COMMON_COA_ID.COMMON_COA_ID = RESULT_MASTER.COMMON_COA_ID
    AND RESULT_SCENARIO.RESULT_SYS_ID = RESULT_MASTER.RESULT_SYS_ID
    AND RESULT_SCENARIO.SCENARIO_NUM = RESULT_MASTER.SCENARIO_NUM
```

```
      AND RESULT_BUCKET.RESULT_SYS_ID = RESULT_MASTER.RESULT_SYS_ID
      AND RESULT_BUCKET.DGAP_SCENARIO_NUM = RESULT_MASTER.SCENARIO_NUM
      AND Result_scenario.description =: Res_Sce_Des
      AND RESULT_BUCKET.FROM_DATE_001-1 =: date2
GROUP BY RESULT_BUCKET.FROM_DATE_001-1, RESULT_BUCKET.FROM_DATE_001-1,
      OFS_RPT_HIER_COMMON_COA_ID.LEAF_DESCRIPTION, RESULT_SCENARIO.DESCRIPTION
```

22.4 Detail/Summary Cashflow Report

The Detail/Summary Cash Flow report shows detailed information about payment,
prepayment, and repricing activity. It enables the user to report on repricing activity
and to take prepayment into account when reviewing cash flow.

- File name = rmcashfl.rdf

SQL Documentation

The Risk Manager table, OFS_RPT_RES_DTL_1000018095_C, contains all financial
elements in their own columns. This series of queries pivots the data to produce all
financial elements within the same column and a financial element description
within a second column. This is done to enable Oracle Reports to create a matrix out
of the resulting data. Additional columns have been added for sorting, highlighting,
and value formatting. The data within the table is organized as follows: Start_Date,
End_Date, Beg_Bal, End_Bal, 01-JAN-1998, 31-JAN-1998, 100, 90. Following the
query, the data is presented in the following manner: Start_Date, End_Date, Row_
Description, Value, 01-JAN-1998, 31-JAN-1998, Beginning Balance, 100,
01-JAN-1998, 31-JAN-1998, End Balance, 90.

```
-- Query 01 --
SELECT 10 "Sort", 1 "GroupFlag", Null "FormatString", Start_Date "Start Date",
    End_Date "End Date", 'Beginning Balance' "Row Description",
    To_Number(Null) "Value"
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE, Result_Scenario
    Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date AND :Ending_Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION
```

```
-- Query 02 --
SELECT 20, 0, '$', Start_Date, End_Date, ' ' || Display_Name, SUM(Beg_Bal)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'BEG_BAL'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 03 --
SELECT 30, 0, '%', Start_Date, End_Date, '  Beginning Rate',
    DECODE(SUM(Beg_Bal),0,0,SUM(Beg_Net_ Rate)/SUM(Beg_Bal))*100
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE, Result_Scenario
    Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 04 --
SELECT 40, 1, Null, Start_Date, End_Date, 'Repricing Activity',
    To_Number(Null)
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE, Result_Scenario
    Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
     Result_Scenario.Description, Start_Date, End_Date
UNION
```

```
-- Query 05 --
SELECT 50, 0, '$', Start_Date, End_Date, ' ' || Display_Name,
    SUM(Reprice_Bal)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'REPRICE_BAL'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION
-- Query 06 --
SELECT 60, 0, '%', Start_Date, End_Date, '  Before Reprice Rate',
    DECODE(SUM(Reprice_Bal), 0, 0, SUM(Bfr_Reprice_Net_Rate)/SUM(Reprice_
    Bal))*100
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE, Result_Scenario
    Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION
-- Query 07 --
SELECT 70, 0, '$', Start_Date, End_Date, ' ' || Display_Name,
    SUM(Reprice_Bal_End)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'REPRICE_BAL_END'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_Date
```

```
            GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                Result_Scenario.Description, Start_Date, End_Date, Display_Name
            UNION
            -- Query 08 --
            SELECT 80, 0, '%', Start_Date, End_Date, '  After Reprice Rate',
                DECODE(SUM(Reprice_Bal_End), 0, 0, SUM(Aft_Reprice_Net_Rate)/SUM(Reprice_
                Bal_ End))*100
            FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                Result_Scenario Result_Scenario
            WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                AND leaf_description = :Rpt_Hier_Leaf_Description
                AND Result_Scenario.Description = :Res_Scen_Description
                AND Start_Date Between :Starting_Date and :Ending_ Date
            GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                Result_Scenario.Description, Start_Date, End_Date
            UNION

            -- Query 09 --
            SELECT 90, 1, Null, Start_Date, End_Date, 'Cash Flow Activity',
                To_Number(Null)
            FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                  Result_Scenario Result_Scenario
            WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                AND leaf_description = :Rpt_Hier_Leaf_Description
                AND Result_Scenario.Description = :Res_Scen_Description
                AND Start_Date Between :Starting_Date and :Ending_ Date
            GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                Result_Scenario.Description, Start_Date, End_Date
            UNION

            -- Query 10 --
            SELECT 100, 0 , '$', Start_Date, End_Date, '  Sched Principal Runoff',
                SUM(PAYMENT_RUNOFF_POS+PAYMENT_RUNOFF_NEG)/1000
            FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                Result_Scenario Result_Scenario
            WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                AND leaf_description = :Rpt_Hier_Leaf_Description
                AND Result_Scenario.Description = :Res_Scen_Description
                AND Start_Date Between :Starting_Date and :Ending_ Date
```

```
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 11 --
SELECT 110, 0, '$', Start_Date, End_Date, '  Principal at Maturity',
    SUM(MAT_RUNOFF_POS+MAT_RUNOFF_NEG)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION
-- Query 12 --
SELECT 120, 0 , '$', Start_Date, End_Date, '  Prepayments',
    SUM(PREPAY_RUNOFF_POS+PREPAY_RUNOFF_NEG)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 13 --
SELECT 130, 0 , '$', Start_Date, End_Date, '  ' || Display_Name,
    SUM(BAL_BEFORE_PREPAY)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'BAL_BEFORE_PREPAY'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
```

```
                AND Start_Date Between :Starting_Date and :Ending_ Date
        GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
            Result_Scenario.Description, Start_Date, End_Date, Display_Name
        UNION
        -- Query 14 --
        SELECT 140, 0, '%', Start_Date, End_Date, '  Annual Prepay Date',
            DECODE(SUM(Bal_Before_Prepay), 0, 0, SUM(Ann_Prepay_Rate)/SUM(Bal_Before_
            Prepay))*100
        FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
            OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
            Result_Scenario Result_Scenario
        WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
            AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
            AND leaf_description = :Rpt_Hier_Leaf_Description
            AND Start_Date Between :Starting_Date and :Ending_ Date
            AND Result_Scenario.Description = :Res_Scen_Description
        GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
            Result_Scenario.Description, Start_Date, End_Date
        UNION

        -- Query 15 --
        SELECT 150, 0 , '$', Start_Date, End_Date, '  Tot Principal Runoff',
            SUM(TOT_RUNOFF_POS+TOT_RUNOFF_NEG)/1000
        FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
            OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
            Result_Scenario Result_Scenario
        WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
            AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
            AND leaf_description = :Rpt_Hier_Leaf_Description
            AND Result_Scenario.Description = :Res_Scen_Description
            AND Start_Date Between :Starting_Date and :Ending_ Date
        GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
            Result_Scenario.Description, Start_Date, End_Date
        UNION

        -- Query 16 --
        SELECT 160, 0, '$', Start_Date, End_Date, '  ' || Display_Name,
            SUM(Int_Cash_Flow)/1000
        FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
            OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
            Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
        WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
            AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
            AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
            AND ALL_TAB_COLUMNS.Column_Name = 'INT_CASH_FLOW'
```

```
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 17 --
SELECT 170, 0 , '$', Start_Date, End_Date, ' ' || Display_Name,
    SUM(Int_Credited)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'INT_CREDITED'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 18 --
SELECT 180, 0 , '$', Start_Date, End_Date, '  Total Cash Flow',
    SUM(TOT_RUNOFF_POS+TOT_RUNOFF_NEG+INT_CASH_FLOW- INT_CREDITED)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 19 --
SELECT 190, 1, Null , Start_Date, End_Date, 'Accrual Income',
    To_Number(Null)
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
```

```
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 20 --
SELECT 200, 0, '$', Start_Date, End_Date, ' ' || Display_Name,
    SUM(Int_Accrued)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'INT_ACCRUED'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 21 --
SELECT 210, 0, '$', Start_Date, End_Date, ' ' || Display_Name,
    SUM(Deferred_Runoff)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'DEFERRED_RUNOFF'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 22 --
SELECT 220, 1 , Null , Start_Date, End_Date, 'New Business Activity',
```

```
        To_Number(Null)
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE, Result_Scenario
    Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION
-- Query 23 --
SELECT 230, 0, '$', Start_Date, End_Date, '  Originations',
    SUM(New_Add_Bal)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 24 --
SELECT 240, 0, '%', Start_Date, End_Date, '  Rate on Originations',
    DECODE(SUM(New_Add_Bal), 0, 0, SUM(New_Add_Net_Rate)/SUM(New_Add_Bal))*100
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 25 --
SELECT 250, 0, '$', Start_Date, End_Date, '  Rolling Balances',
    SUM(Roll_Add_Bal)/1000
```

```
                FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                    Result_Scenario Result_Scenario
                WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                    AND leaf_description = :Rpt_Hier_Leaf_Description
                    AND Result_Scenario.Description = :Res_Scen_Description
                    AND Start_Date Between :Starting_Date and :Ending_ Date
                GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                    Result_Scenario.Description, Start_Date, End_Date
                UNION

                -- Query 26 --
                SELECT 260, 0, '%', Start_Date, End_Date, '  Rate on Rolling Balances',
                    DECODE(SUM(Roll_Add_Bal), 0, 0, SUM(Roll_Add_Net_Rate)/SUM(Roll_Add_
                    Bal))*100
                FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                    Result_Scenario Result_Scenario
                WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                    AND leaf_description = :Rpt_Hier_Leaf_Description
                    AND Result_Scenario.Description = :Res_Scen_Description
                    AND Start_Date Between :Starting_Date and :Ending_ Date
                GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                    Result_Scenario.Description, Start_Date, End_Date
                UNION

                -- Query 27 --
                SELECT 270, 0, '$', Start_Date, End_Date, '  Total New Business Activity',
                    SUM(New_Add_Bal+Roll_Add_Bal)/1000
                FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
                    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
                    Result_Scenario Result_Scenario
                WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
                    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
                    AND leaf_description = :Rpt_Hier_Leaf_Description
                    AND Result_Scenario.Description = :Res_Scen_Description
                    AND Start_Date Between :Starting_Date and :Ending_ Date
                GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
                    Result_Scenario.Description, Start_Date, End_Date
                UNION

                -- Query 28 --
                SELECT 280, 1 , Null , Start_Date, End_Date, 'Ending Balance',
```

```
       To_Number(Null)
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION

-- Query 29 --
SELECT 290, 0, '$', Start_Date, End_Date, '  ' || Display_Name,
    SUM(End_Bal)/1000
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
    AND ALL_TAB_COLUMNS.Column_Name = 'END_BAL'
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date, Display_Name
UNION

-- Query 30 --
SELECT 300, 0, '%', Start_Date, End_Date, '  Ending Rate',
    DECODE(SUM(End_Bal), 0, 0, SUM(End_Net_Rate)/SUM(End_Bal))*100
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
UNION
-- Query 31 --
```

```
          SELECT 310, 0, '%', Start_Date, End_Date, '  Fully Indexed Rate',
              DECODE(SUM(End_Bal), 0, 0, SUM(Fully_Ind_Net_Rate)/SUM(End_Bal))*100
          FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
              OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
              Result_Scenario Result_Scenario
          WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
              AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
              AND leaf_description = :Rpt_Hier_Leaf_Description
              AND Result_Scenario.Description = :Res_Scen_Description
              AND Start_Date Between :Starting_Date and :Ending_ Date
          GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
              Result_Scenario.Description, Start_Date, End_Date
          UNION

          -- Query 32 --
          SELECT 320, 0, '$', Start_Date, End_Date, '  ' || Display_Name,
              SUM(Avg_Bal)/1000
          FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
              OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
              Result_Scenario Result_Scenario, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
          WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
              AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
              AND ALL_TAB_COLUMNS.Table_Name = 'OFS_RPT_RES_DTL_ 1000018095_C'
              AND ALL_TAB_COLUMNS.Column_Name = 'AVG_BAL'
              AND leaf_description = :Rpt_Hier_Leaf_Description
              AND Result_Scenario.Description = :Res_Scen_Description
              AND Start_Date Between :Starting_Date and :Ending_ Date
          GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
              Result_Scenario.Description, Start_Date, End_Date, Display_Name
          UNION
          -- Query 33 --
          SELECT 330, 0, '%', Start_Date, End_Date, '  Average Rate',
              DECODE(SUM(Avg_Bal), 0, 0, SUM(Avg_Net_Rate)/SUM(Avg_Bal))*100
          FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
              OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
              Result_Scenario Result_Scenario
          WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
              AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
              AND leaf_description = :Rpt_Hier_Leaf_Description
              AND Result_Scenario.Description = :Res_Scen_Description
              AND Start_Date Between :Starting_Date and :Ending_ Date
              GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
              Result_Scenario.Description, Start_Date, End_Date
          UNION
```

```
-- Query 34 --
SELECT 340, 0, '', Start_Date, End_Date, '  Wtd Avg Rem Months',
    ROUND(DECODE(SUM(End_Bal), 0, 0, SUM(WARM)/SUM(End_Bal)/100)*1000,0)
FROM OFS_RPT_RES_DTL_1000018095_C STD_RPT_RES_DTL_ CASH,
    OFS_RPT_HIER_TM_ACCT_TYPE OFS_RPT_HIER_TM_ACCT_ TYPE,
    Result_Scenario Result_Scenario
WHERE OFS_RPT_HIER_TM_ACCT_TYPE.TMCOA_Id = STD_RPT_RES_DTL_CASH.TMCOA_Id
    AND Result_Scenario.Scenario_Num = STD_RPT_RES_DTL_CASH.Scenario_Num
    AND leaf_description = :Rpt_Hier_Leaf_Description
    AND Result_Scenario.Description = :Res_Scen_Description
    AND Start_Date Between :Starting_Date and :Ending_ Date
GROUP BY OFS_RPT_HIER_TM_ACCT_TYPE.Leaf_Description,
    Result_Scenario.Description, Start_Date, End_Date
-- Sort Order --
Order by 4,1 ;
```

22.5 Gap Detail Report

This report shows account level gap runoff information for a single start date.

- File name = rmgapdet.rdf

SQL Documentation

```
SELECT OFS_RPT_RES_DTL_1000018095_G.DYNAMIC_GAP_DATE AS E151625,
    OFS_RPT_RES_DTL_1000018095_G.START_DATE AS E151626,
    OFS_RPT_RES_DTL_1000018095_G.END_DATE AS E151627,
    OFSA_RESULT_SCENARIO.DESCRIPTION AS E152500, SUM(OFS_RPT_RES_DTL_1000018095_
    G.GAP_DEFERRED_RUNOFF) AS E151635_SUM, SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_
    INT_CREDITED) AS E151641_SUM, SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_INT_CF_
    NET) AS E151637_SUM, SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_RUNOFF) AS
    E151632_SUM, SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_PRIN_RUNOFF) AS E151633_
    SUM, SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_REPRICE_RUNOFF) AS E151634_SUM,
    DECODE(SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_RUNOFF),0,0,SUM(OFS_RPT_RES_DTL_
    1000018095_G.GAP_RUNOFF_TERM)/SUM(OFS_RPT_RES_DTL_1000018095_G.GAP_RUNOFF))
    AS E_379, SUM(NULL) AS E_374, SUM(NULL) AS E_371, SUM(DECODE(OFSA_RESULT_
    MASTER.FINANCIAL_ROLLUP,300,OFS_RPT_RES_DTL_1000018095_G.GAP_ACCRUED_INT_
    NET*-1,310,OFS_RPT_RES_DTL_1000018095_G.GAP_ACCRUED_INT_NET*-1,620,OFS_RPT_
    RES_DTL_1000018095_G.GAP_ACCRUED_INT_NET*-1,OFS_RPT_RES_DTL_1000018095_
    G.GAP_ACCRUED_INT_NET)) AS E_365, DECODE(SUM(OFS_RPT_RES_DTL_1000018095_
    G.GAP_RUNOFF),0,0,( SUM(( OFS_RPT_RES_DTL_1000018095_G.GAP_RUNOFF_NET_
    RATE*OFS_RPT_RES_DTL_1000018095_G.GAP_RUNOFF ))/SUM(OFS_RPT_RES_DTL_
    1000018095_G.GAP_RUNOFF) )/1) AS E151674
FROM OFSA_RESULT_MASTER OFSA_RESULT_MASTER,
    OFS_RPT_RES_DTL_1000018095_G OFS_RPT_RES_DTL_1000018095_G, OFSA_RESULT_
```

```
                    SCENARIO OFSA_RESULT_SCENARIO
    WHERE ((OFSA_RESULT_MASTER.COMMON_COA_ID = OFS_RPT_RES_DTL_1000018095_G.TMCOA_ID
        AND ( OFSA_RESULT_SCENARIO.SCENARIO_NUM = OFS_RPT_RES_DTL_1000018095_
        G.SCENARIO_NUM ) ) AND ( OFSA_RESULT_SCENARIO.RESULT_SYS_ID = 1000018095 )
    GROUP BY OFS_RPT_RES_DTL_1000018095_G.DYNAMIC_GAP_DATE,
        OFS_RPT_RES_DTL_1000018095_G.START_DATE, OFS_RPT_RES_DTL_1000018095_G.END_
        DATE, OFSA_RESULT_SCENARIO.DESCRIPTION
```

22.6 Scenario Income Statement

The Scenario Income Statement report shows the change in interest income for a multitude of time periods at both the product and scenario level.

- File name = rmincsce.rdf

SQL Documentation

```
SELECT REPORTING_RES_DTL_C.START_DATE AS E152159,
    REPORTING_RES_DTL_C.END_DATE AS E152160, OFSA_RESULT_TYPE_DSC.RESULT_TYPE AS
    E152340, OFS_RPT_HIER_COMMON_COA_ID.LEAF_DESCRIPTION AS E152372, OFSA_
    RESULT_SCENARIO.DESCRIPTION AS E159048, SUM(DECODE(OFSA_RESULT_
    MASTER.FINANCIAL_ROLLUP,300,REPORTING_RES_DTL_C.INT_
    ACCRUED*-1,310,REPORTING_RES_DTL_C.INT_ACCRUED*-1,620,REPORTING_RES_DTL_
    C.INT_ACCRUED*-1,REPORTING_RES_DTL_C.INT_ACCRUED))+SUM(REPORTING_RES_DTL_
    C.NON_INT_INC-REPORTING_RES_DTL_C.NON_INT_EXP+REPORTING_RES_DTL_C.FED_
    TAXES+REPORTING_RES_DTL_C.LOCAL_TAXES+REPORTING_RES_DTL_C.DIVIDENDS) AS E_11
FROM OFSA_RESULT_MASTER OFSA_RESULT_MASTER, REPORTING_RES_DTL_C
    REPORTING_RES_DTL_C, OFSA_RESULT_TYPE_DSC OFSA_RESULT_TYPE_DSC, OFS_RPT_
    HIER_COMMON_COA_ID OFS_RPT_HIER_COMMON_COA_ID, OFSA_RESULT_SCENARIO OFSA_
    RESULT_SCENARIO
WHERE ((OFSA_RESULT_TYPE_DSC.RESULT_TYPE_CD = REPORTING_RES_DTL_C.RESULT_TYPE_
CD)
    AND ( OFS_RPT_HIER_COMMON_COA_ID.COMMON_COA_ID = REPORTING_RES_DTL_C.COMMON_
    COA_ID )
    AND ( OFSA_RESULT_MASTER.COMMON_COA_ID = REPORTING_RES_DTL_C.COMMON_COA_ID )
    AND ( OFSA_RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM))
    AND ( OFSA_RESULT_SCENARIO.RESULT_SYS_ID = 101894 )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :START_DATE AND :END_DATE )
    AND ( OFSA_RESULT_SCENARIO.DESCRIPTION = :DESCRIPTION )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    OFSA_RESULT_TYPE_DSC.RESULT_TYPE, OFS_RPT_HIER_COMMON_COA_ID.LEAF_
    DESCRIPTION, OFSA_RESULT_SCENARIO.DESCRIPTION
```

22.7 Summary Income Statement

The Summary Income Statement is a standard Income Statement, but enables the user to base the statement on scenarios instead of current position.

- File name = rmincsum.rdf

SQL Documentation

```
-- Query 1 --
SELECT 10 "Sort_Order", 0 "GroupFlag", REPORTING_RES_DTL_C.START_DATE as "Start
Date",
    REPORTING_RES_DTL_C.END_DATE as "End Date", ALL_TAB_COLUMNS.COLUMN_NAME as
    "Row Description", Sum(REPORTING_RES_DTL_C.INT_ACCRUED) as "Value"
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C,
    OFSA_RESULT_SCENARIO RESULT_SCENARIO,ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
    Dtl_End_Date )
    AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C' )
    AND( ALL_TAB_COLUMNS.COLUMN_NAME = 'INT_ACCRUED' )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
UNION

-- Query 2 --
SELECT 20, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.DEFERRED_RUNOFF)
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C,
    OFSA_RESULT_SCENARIO RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
    Dtl_End_Date )
    AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C' )
    AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'DEFERRED_RUNOFF' )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
UNION

-- Query 3 --
SELECT 30, 1, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    'Net Interest Accrued', SUM(REPORTING_RES_DTL_C.INT_ACCRUED + REPORTING_RES_
    DTL_C.DEFERRED_RUNOFF)
```

```
         FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
             RESULT_SCENARIO
         WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
             AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
             AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
             Dtl_End_Date )
         GROUP BY REPORTING_RES_DTL_C.START_DATE,REPORTING_RES_DTL_C.END_DATE,
             RESULT_SCENARIO.DESCRIPTION
         UNION

         -- Query 4 --
         SELECT  40, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
             ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.NON_INT_INC)
         FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C,OFSA_RESULT_SCENARIO
             RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
         WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
             AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
             AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
             Dtl_End_Date )
             AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C')
             AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'NON_INT_INC' )
         GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
             RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
         UNION

         -- Query 5 --
         SELECT 50, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
             ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.NON_INT_EXP)
         FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C,
             OFSA_RESULT_SCENARIO RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
         WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
             AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
             AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
             Dtl_End_Date )
             AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C')
             AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'NON_INT_EXP' )
         GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
             RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
         UNION

         -- Query 6 --
         SELECT 60, 1, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
             'Net Non-Interest Income', SUM(REPORTING_RES_DTL_C.NON_INT_INC - REPORTING_
             RES_DTL_C.NON_INT_EXP)
         FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
```

```
                RESULT_SCENARIO
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
    Dtl_End_Date )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    RESULT_SCENARIO.DESCRIPTION
UNION

-- Query 7 --
SELECT 70, 1, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    'Net Income before Tax',SUM( REPORTING_RES_DTL_C.NON_INT_INC + REPORTING_
    RES_DTL_C.NON_INT_EXP )
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
    RESULT_SCENARIO
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
    Dtl_End_Date )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    RESULT_SCENARIO.DESCRIPTION
UNION

-- Query 8 --
SELECT 80, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.LOCAL_TAXES)
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C,OFSA_RESULT_SCENARIO
    RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
    AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
    Dtl_End_Date )
    AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C')
    AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'LOCAL_TAXES' )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
UNION

-- Query 9 --
SELECT 90, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
    ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.FED_TAXES)
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
    RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
    AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
```

```
       AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
       Dtl_End_Date )
       AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C')
       AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'FED_TAXES' )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
       RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
UNION

-- Query 10 --
SELECT 100, 1, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
       'Net Income After Taxes', SUM( REPORTING_RES_DTL_C.NON_INT_INC
       + REPORTING_RES_DTL_C.NON_INT_EXP + REPORTING_RES_DTL_C.LOCAL_TAXES
       + REPORTING_RES_DTL_C.FED_TAXES)
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
       RESULT_SCENARIO
WHERE( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
       AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
       AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
       Dtl_End_Date )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
       RESULT_SCENARIO.DESCRIPTION
UNION

-- Query 11 --
SELECT 110, 0, REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
       ALL_TAB_COLUMNS.COLUMN_NAME, SUM(REPORTING_RES_DTL_C.DIVIDENDS)
FROM REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_SCENARIO
       RESULT_SCENARIO, ALL_TAB_COLUMNS ALL_TAB_COLUMNS
WHERE ( ( RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM ) )
       AND ( RESULT_SCENARIO.DESCRIPTION = :Res_Scn_Desc )
       AND ( REPORTING_RES_DTL_C.START_DATE BETWEEN :Res_Dtl_Start_Date and :Res_
       Dtl_End_Date )
       AND ( ALL_TAB_COLUMNS.TABLE_NAME = 'REPORTING_RES_DTL_C')
       AND ( ALL_TAB_COLUMNS.COLUMN_NAME = 'DIVIDENDS' )
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE,
       RESULT_SCENARIO.DESCRIPTION, ALL_TAB_COLUMNS.COLUMN_NAME
Order by 3, 1
```

23

Oracle Transfer Pricing Reports

The Oracle Transfer Pricing reports track the profitability of different operations and product lines by reporting the transfer rate on financial products. Using the same sophisticated stochastic techniques available in Oracle Risk Manager, the Transfer Pricing processing engine calculates and outputs the cost of optionality. Based on the spread over transfer rate, this calculation quanitfies the cost of optionality for a single instrument at the time of funding, thus isolating the line officers from the variability of funding costs.

Currently, only one report is available for Transfer Pricing: the Transfer Pricing Income Statement report. This report shows the forecasted contribution to the funding center by assets & liabilities. Also, it reports the net income generated by the funding center by month and scenario.

Report Name
Transfer Pricing Income Statement

File Name
rminctp.rdf

SQL Documentation
The following query was copied and pasted from a Discoverer workbook, then modified for reporting:

```
SELECT REPORTING_RES_DTL_C.START_DATE AS E152159,
    REPORTING_RES_DTL_C.END_DATE AS E152160,
    OFSA_RESULT_TYPE_DSC.RESULT_TYPE AS E152340,
    OFSA_RESULT_SCENARIO.DESCRIPTION AS E159048,
    ((SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_ROLLUP,100,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,110,REPORTING_RES_DTL_C.CHARGE_CREDIT,610,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,0)) ) )-( ( SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,300,REPORTING_RES_DTL_C.CHARGE_CREDIT,310,REPORTING_RES_DTL_C.CHARGE_
    CREDIT,620,REPORTING_RES_DTL_C.CHARGE_CREDIT,0)) ) ) AS E_824, (
    SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_ROLLUP,300,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,310,REPORTING_RES_DTL_C.CHARGE_CREDIT,620,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,0)) ) AS E_822, ( SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,100,REPORTING_RES_DTL_C.CHARGE_CREDIT,110,REPORTING_RES_DTL_C.CHARGE_
    CREDIT,610,REPORTING_RES_DTL_C.CHARGE_CREDIT,0)) ) AS E_820, (
    SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_ROLLUP,300,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,310,REPORTING_RES_DTL_C.CHARGE_CREDIT,620,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,0)) )-( SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,300,REPORTING_RES_DTL_C.INT_ACCRUED,310,REPORTING_RES_DTL_C.INT_
    ACCRUED,620,REPORTING_RES_DTL_C.INT_ACCRUED,0)) ) AS E_816, SUM(DECODE(OFSA_
    RESULT_MASTER.FINANCIAL_ROLLUP,300,REPORTING_RES_DTL_C.INT_
    ACCRUED,310,REPORTING_RES_DTL_C.INT_ACCRUED,620,REPORTING_RES_DTL_C.INT_
    ACCRUED,0)) AS E_811, SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,300,REPORTING_RES_DTL_C.CHARGE_CREDIT,310,REPORTING_RES_DTL_C.CHARGE_
    CREDIT,620,REPORTING_RES_DTL_C.CHARGE_CREDIT,0)) AS E_806, SUM(NULL) AS E_
    803, ( SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_ROLLUP,100,REPORTING_RES_DTL_
    C.INT_ACCRUED,110,REPORTING_RES_DTL_C.INT_ACCRUED,610,REPORTING_RES_DTL_
    C.INT_ACCRUED,0)) )-( SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,100,REPORTING_RES_DTL_C.CHARGE_CREDIT,110,REPORTING_RES_DTL_C.CHARGE_
    CREDIT,610,REPORTING_RES_DTL_C.CHARGE_CREDIT,0)) ) AS E_793,
    SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_ROLLUP,100,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,110,REPORTING_RES_DTL_C.CHARGE_CREDIT,610,REPORTING_RES_DTL_
    C.CHARGE_CREDIT,0)) AS E_788, SUM(DECODE(OFSA_RESULT_MASTER.FINANCIAL_
    ROLLUP,100,REPORTING_RES_DTL_C.INT_ACCRUED,110,REPORTING_RES_DTL_C.INT_
    ACCRUED,610,REPORTING_RES_DTL_C.INT_ACCRUED,0)) AS E_783
FROM OFSA_RESULT_MASTER OFSA_RESULT_MASTER,
    REPORTING_RES_DTL_C REPORTING_RES_DTL_C, OFSA_RESULT_TYPE_DSC OFSA_RESULT_
    TYPE_DSC, OFSA_RESULT_SCENARIO OFSA_RESULT_SCENARIO
WHERE OFSA_RESULT_TYPE_DSC.RESULT_TYPE_CD = REPORTING_RES_DTL_C.RESULT_TYPE_CD
    AND OFSA_RESULT_MASTER.COMMON_COA_ID = REPORTING_RES_DTL_C.COMMON_COA_ID
    AND OFSA_RESULT_SCENARIO.SCENARIO_NUM = REPORTING_RES_DTL_C.SCENARIO_NUM
    AND OFSA_RESULT_SCENARIO.RESULT_SYS_ID = 101894
    AND OFSA_RESULT_SCENARIO.DESCRIPTION =: RES_DESC
    AND OFSA_RESULT_TYPE_DSC.RESULT_TYPE =: RES_TYPE
GROUP BY REPORTING_RES_DTL_C.START_DATE, REPORTING_RES_DTL_C.END_DATE, OFSA_
RESULT_TYPE_DSC.RESULT_TYPE, OFSA_RESULT_SCENARIO.DESCRIPTION
```

Index