

Oracle® Transfer Pricing

User Guide

Release 12

Part No. B26000-01

December 2006

Oracle Transfer Pricing User Guide, Release 12

Part No. B26000-01

Copyright © 2006, Oracle. All rights reserved.

Primary Author: Vijay Tiwary

Contributing Author: Mayur Polepalli

Contributor: Amit Budhiraja, Lokesh Garg, Jack Hickox, Angel Lafont, Essan Ni, Christopher Spofford

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments

Preface

1 Introduction to Oracle Transfer Pricing

Overview of Oracle Transfer Pricing.....	1-1
Oracle Transfer Pricing and Other Oracle Financial Services Applications.....	1-2
Oracle Transfer Pricing Integrations.....	1-3

2 The Oracle Transfer Pricing Process

Overview of the Oracle Transfer Pricing Process.....	2-1
Setting Transfer Pricing Rules.....	2-3
Transfer Pricing Methodologies and Rules.....	2-4
Duration.....	2-5
Weighted Average Cash Flow.....	2-6
Zero Coupon Pricing.....	2-7
Moving Averages.....	2-9
Straight Term.....	2-10
Spread from Interest Rate Code.....	2-11
Spread from Note Rate.....	2-11
Redemption Curve.....	2-12
Unpriced Account.....	2-13
Transfer Pricing Methods and the Mid-Period Repricing Option.....	2-14
Defining Transfer Pricing Methodologies Using Node Level Assumptions.....	2-19
Associating Conditional Assumptions with Transfer Pricing Rules.....	2-21
Availability of Conditional Assumptions.....	2-22

Structure of Conditional Assumptions.....	2-24
Setting Prepayment Rules.....	2-27
Prepayment Methodologies and Rules.....	2-27
Constant Prepayment Method.....	2-28
Prepayment Table Method.....	2-28
Arctangent Calculation Method.....	2-31
Associating Node Level and Conditional Assumptions with Prepayment Rules.....	2-35
Setting and Executing the Transfer Pricing Process Rule.....	2-36
Transfer Pricing Process Rule and Option Cost Parameters.....	2-37
Transfer Pricing Process Rule and Rate Index Rules.....	2-38
Transfer Pricing Process Rule and Propagation Pattern.....	2-39
Transfer Pricing Process Rule and Audit Options.....	2-39
Data Inspector Results.....	2-40
Data Verification.....	2-42
Transfer Pricing Process Rule and Calculation Modes.....	2-42
Transfer Pricing Process Rule and Migration Options.....	2-43
Defining Payment Patterns.....	2-44
Payment Pattern Structure.....	2-44
Payment Events.....	2-45
Absolute Payment Patterns.....	2-47
Relative Payment Patterns.....	2-48
Split Payment Patterns.....	2-48
Defining Repricing Patterns.....	2-48
User Defined Repricing Pattern.....	2-49
User Defined Repricing Event.....	2-49
Absolute Repricing Pattern.....	2-51
Relative Repricing Pattern.....	2-52
Performing Cash Flow Edits.....	2-52
Creating Interest Rate Codes.....	2-53
Interest Rate Codes and Rate Lookups.....	2-53
Accessing Transfer Pricing Detail Cash Flow Results for Audit Purposes.....	2-55
Analyzing Results.....	2-57
Reviewing Processing Errors.....	2-58
Reprocessing Erroneous Accounts.....	2-59
Reconciling the Data.....	2-59

3 Common Rule Management Tasks

Overview of Common Rule Management Tasks.....	3-1
The Rule Home Page.....	3-2
Searching for Rules.....	3-5

Creating Rules.....	3-6
Viewing and Updating Rules.....	3-7
Duplicating Rules.....	3-7
Deleting Rules.....	3-8
Extracting and Loading Rules.....	3-9
4 Working with Rule Approval Status	
About Rule Approval and Production Data Sets.....	4-1
The Rule Approval Process.....	4-2
The Rule Deletion Process.....	4-2
5 Managing Currency Rates	
Overview of Currency Rates Management.....	5-1
Overview of Multi-Currency Accounting.....	5-3
Overview.....	5-3
Currencies.....	5-9
Defining Currencies.....	5-9
Conversion Rates.....	5-11
Defining Conversion Rate Types.....	5-11
Entering Daily Rates.....	5-13
Loading Daily Rates Automatically.....	5-18
Entering Historical Rates.....	5-23
Revaluing Balances.....	5-27
Translating Balances.....	5-36
Notes on Translation with Historical Rates and Amounts.....	5-49
Notes on Translating Owners' Equity Accounts.....	5-50
Notes on Translating Revenue/Expense Accounts.....	5-51
Notes on Translating Average Balances.....	5-53
Currency Rates Manager.....	5-57
Cross Rates.....	5-57
Using the Daily Rates Spreadsheet Interface.....	5-61
Using the Historical Rates Spreadsheet Interface.....	5-62
6 Cash Flow Edits	
Overview of Cash Flow Edit Rules.....	6-1
Creating Cash Flow Edit Rules.....	6-2
Executing Cash Flow Edit Rules.....	6-4

7	User Defined Payment Pattern	
	Overview of User Defined Payment Patterns.....	7-1
	Searching for Payment Patterns.....	7-2
	Creating Payment Patterns.....	7-3
	Defining Absolute Payment Patterns.....	7-4
	Defining Relative Payment Patterns	7-7
	Defining Split Payment Patterns	7-10
8	User Defined Repricing Pattern	
	Overview of Repricing Patterns.....	8-1
	Searching for Repricing Patterns.....	8-2
	Creating Repricing Patterns.....	8-3
	Defining Absolute Repricing Patterns.....	8-4
	Defining Relative Repricing Patterns.....	8-7
9	Interest Rate Codes	
	Overview of Interest Rate Codes.....	9-1
	Searching for Interest Rate Codes.....	9-2
	Creating Interest Rate Codes.....	9-3
	Loading Data.....	9-5
	Loading Data Using Oracle Transfer Pricing User Interface.....	9-6
	Loading Data Using Web ADI	9-8
10	Rate Index Rules	
	Overview of Rate Index Rules.....	10-1
	Defining Rate Indexes.....	10-2
11	Conditional Assumptions	
	Overview of Conditional Assumptions.....	11-1
	Creating Conditional Assumptions.....	11-1
	Inserting a Method.....	11-9
	Inserting Logic into a Block.....	11-11
	Updating a Clause.....	11-12
12	Transfer Pricing Rules	
	Overview of Transfer Pricing Rules.....	12-1
	Creating Transfer Pricing Rules.....	12-2
	Defining Transfer Pricing Methodologies.....	12-3

Defining the Redemption Curve Methodology.....	12-9
Defining the Unpriced Account Methodology	12-10
13 Prepayment Rules	
Overview of Prepayment Rules	13-1
Creating Prepayment Rules	13-2
Defining Prepayment Methodologies	13-3
Defining the Constant Prepayment Method.....	13-6
Defining the Prepayment Table Method.....	13-8
Defining the Arctangent Calculation Method.....	13-9
14 Prepayment Table Rules	
Overview of Prepayment Table Rules	14-1
Creating Prepayment Table Rules	14-2
Updating Prepayment Tables	14-6
Updating Prepayment Rates in a Prepayment Table	14-7
15 Propagation Pattern	
Overview of the Propagation Pattern	15-1
Defining the Propagation Pattern	15-1
Propagating Transfer Pricing Results	15-3
16 Transfer Pricing Process Rules	
Overview of Transfer Pricing Process Rules	16-1
Creating a Transfer Pricing Process Rule	16-2
Executing a Transfer Pricing Process Rule	16-9
17 Oracle Transfer Pricing Reports	
Overview of Oracle Transfer Pricing Reports	17-1
Generating and Viewing Oracle Transfer Pricing Reports	17-2
Oracle Transfer Pricing Reports Common Concepts.....	17-3
STD - FTP Interest Margin Account.....	17-5
STD - FTP Interest Margin Account (Org/Product, Summary).....	17-6
STD - FTP Margin Stratification.....	17-6
A Standard Navigation Paths	
Standard Navigation Paths	A-1

Glossary

Index

Send Us Your Comments

Oracle Transfer Pricing User Guide, Release 12

Part No. B26000-01

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document. Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Applications Release Online Documentation CD available on Oracle MetaLink and www.oracle.com. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: appsdoc_us@oracle.com

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at www.oracle.com.

Preface

Intended Audience

Welcome to Release 12 of the *Oracle Transfer Pricing User Guide*.

This guide assumes you have a working knowledge of the following:

- The principles and customary practices of your business area.
- Computer desktop application usage and terminology

If you have never used Oracle Applications, we suggest you attend one or more of the Oracle Applications training classes available through Oracle University.

See Related Information Sources on page xiii for more Oracle Applications product information.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site

at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

1 Introduction to Oracle Transfer Pricing

This chapter provides an introduction to Oracle Transfer Pricing and discusses its place in the Oracle Financial Services (OFS) group of applications.

2 The Oracle Transfer Pricing Process

This chapter describes the set of steps that you need to follow to transfer price your balance sheet using Oracle Transfer Pricing.

3 Common Rule Management Tasks

This chapter focuses on the rule management tasks that are common across all rules in this application.

4 Working with Rule Approval Status

This chapter discusses the rule approval process and the procedure for managing approved rules.

5 Managing Currency Rates

This chapter discusses the process for managing currency rate information, including creating daily, historical, and cross rates. The chapter also describes how to upload daily rates and upload and download historical rates from spreadsheet-based applications to Oracle Transfer Pricing.

6 Cash Flow Edits

This chapter discusses the procedure for validating and cleansing your Account table data before you process it to generate transfer pricing results.

7 User Defined Payment Pattern

This chapter describes the procedure for capturing instrument payment patterns that are too complex to be accommodated in the standard fields of Account tables.

8 User Defined Repricing Pattern

This chapter discusses the procedure for working with and managing user defined repricing patterns.

9 Interest Rate Codes

This chapter discusses the procedure for working with and managing interest rate codes and loading related data such as historical rates and term structure parameters.

10 Rate Index Rules

This chapter describes the steps you need to take to work with and manage the Rate Index Rules.

11 Conditional Assumptions

This chapter discusses the procedure for assigning transfer pricing and prepayment methodologies to product-currency combinations using conditional logic.

12 Transfer Pricing Rules

This chapter describes the procedure for working with and managing Transfer Pricing rules.

13 Prepayment Rules

This chapter describes the procedure for working with and managing Prepayment rules.

14 Prepayment Table Rules

This chapter describes the procedure to build prepayment tables using Prepayment Table Rules.

15 Propagation Pattern

This chapter describes the procedure for defining the propagation pattern.

16 Transfer Pricing Process Rules

This chapter discusses the procedure for working with and managing the Transfer Pricing Process rules.

17 Oracle Transfer Pricing Reports

This chapter describes the Oracle Transfer Pricing reports and the procedure for generating and viewing them.

A Standard Navigation Paths

This appendix gives you information to navigate through the pages referred to in this guide.

Glossary

Related Information Sources

This document is included on the Oracle Applications Document Library, which is supplied in the Release 12 DVD Pack. You can download soft-copy documentation as PDF files from the Oracle Technology Network at <http://otn.oracle.com/documentation>, or you can purchase hard-copy documentation from the Oracle Store at <http://oraclestore.oracle.com>. The Oracle E-Business Suite Documentation Library Release 12 contains the latest information, including any documents that have changed significantly between releases. If substantial changes to this book are necessary, a revised version will be made available on the online documentation CD on Oracle *MetaLink*.

If this guide refers you to other Oracle Applications documentation, use only the Release 12 versions of those guides.

For a full list of documentation resources for Oracle Applications Release 12, see Oracle Applications Documentation Resources, Release 12, Oracle*MetaLink* Document 394692.1.

Online Documentation

All Oracle Applications documentation is available online (HTML or PDF).

- **PDF** - PDF documentation is available for download from the Oracle Technology Network at <http://otn.oracle.com/documentation>.
- **Online Help** - Online help patches (HTML) are available on Oracle*MetaLink*.
- **Oracle MetaLink Knowledge Browser** - The Oracle*MetaLink* Knowledge Browser lets you browse the knowledge base, from a single product page, to find all documents for that product area. Use the Knowledge Browser to search for release-specific information, such as FAQs, recent patches, alerts, white papers, troubleshooting tips, and other archived documents.
- **Oracle eBusiness Suite Electronic Technical Reference Manuals** - Each Electronic Technical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for a specific Oracle Applications product. This information helps you convert data from your existing applications and integrate Oracle Applications data with non-Oracle applications, and write custom reports for Oracle Applications products. Oracle eTRM is available on Oracle*MetaLink*.

Related Guides

You should have the following related books on hand. Depending on the requirements of your particular installation, you may also need additional manuals or guides.

Oracle Applications Installation Guide: Using Rapid Install:

This guide provides information about using the Rapid Install utility to install Oracle Applications Release 12, or as a part of an upgrade from Release 11i to Release 12. Discusses Standard and Express installations, fresh or Vision Demo database installations, as well as techstack and product upgrades.

Oracle Applications Maintenance Procedures:

This guide describes how to use AD maintenance utilities to complete tasks such as compiling invalid objects, managing parallel processing jobs, and maintaining snapshot information. Part of Maintaining Oracle Applications, a 3-book set that also includes Oracle Applications Patching Procedures and Oracle Applications Maintenance Utilities.

Oracle Applications Maintenance Utilities:

This guide describes how to run utilities, such as AD Administration and AD

Controller, used to maintain the Oracle Applications file system and database. Outlines the actions performed by these utilities, such as monitoring parallel processes, generating Applications files, and maintaining Applications database entities. Part of Maintaining Oracle Applications, a 3-book set that also includes Oracle Applications Patching Procedures and Oracle Applications Maintenance Procedures.

Oracle Applications Patching Procedures:

This guide describes how to patch the Oracle Applications file system and database using AutoPatch, and how to use other patching-related tools like AD Merge Patch, OAM Patch Wizard, and OAM Registered Flagged Files. Describes patch types and structure, and outlines some of the most commonly used patching procedures. Part of Maintaining Oracle Applications, a 3-book set that also includes Oracle Applications Maintenance Utilities and Oracle Applications Maintenance Procedures.

Oracle Applications Upgrade Guide: Release 11i to Release 12:

This guide provides information for DBAs and Applications Specialists who are responsible for upgrading a Release 11i Oracle Applications system (techstack and products) to Release 12. In addition to information about applying the upgrade driver, it outlines pre-upgrade steps and post-upgrade steps, and provides descriptions of product-specific functional changes and suggestions for verifying the upgrade and reducing downtime.

Oracle Alert User's Guide:

This guide explains how to define periodic and event alerts to monitor the status of your Oracle Applications data.

Oracle Application Framework Developer's Guide:

This guide contains the coding standards followed by the Oracle Applications development staff to produce applications built with Oracle Application Framework. This guide is available in PDF format on *OracleMetaLink* and as online documentation in JDeveloper 10g with Oracle Application Extension.

Oracle Application Framework Personalization Guide:

This guide covers the design-time and run-time aspects of personalizing applications built with Oracle Application Framework.

Oracle Applications Concepts:

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12, or contemplating significant changes to a configuration. After describing the Oracle Applications architecture and technology stack, it focuses on strategic topics, giving a broad outline of the actions needed to achieve a particular goal, plus the installation and configuration choices that may be available.

Oracle Applications Developer's Guide:

This guide contains the coding standards followed by the Oracle Applications development staff. It describes the Oracle Application Object Library components needed to implement the Oracle Applications user interface described in the *Oracle*

Applications User Interface Standards for Forms-Based Products. It also provides information to help you build your custom Oracle Forms Developer forms so that they integrate with Oracle Applications.

Oracle Applications Flexfields Guide:

This guide provides flexfields planning, setup, and reference information for the Oracle Applications implementation team, as well as for users responsible for the ongoing maintenance of Oracle Applications product data. This guide also provides information on creating custom reports on flexfields data.

Oracle Applications System Administrator's Guide Documentation Set:

This documentation set provides planning and reference information for the Oracle Applications System Administrator. *Oracle Applications System Administrator's Guide - Configuration* contains information on system configuration steps, including defining concurrent programs and managers, enabling Oracle Applications Manager features, and setting up printers and online help. *Oracle Applications System Administrator's Guide - Maintenance* provides information for frequent tasks such as monitoring your system with Oracle Applications Manager, managing concurrent managers and reports, using diagnostic utilities, managing profile options, and using alerts. *Oracle Applications System Administrator's Guide - Security* describes User Management, data security, function security, auditing, and security configurations.

Oracle Applications User's Guide:

This guide explains how to navigate, enter data, query, and run reports using the user interface (UI) of Oracle Applications. This guide also includes information on setting user profiles, as well as running and reviewing concurrent requests.

Oracle Integration Repository User's Guide:

This guide covers the employment of Oracle Integration Repository in researching and deploying business interfaces to produce integrations between applications.

Oracle Web Applications Desktop Integrator Implementation and Administration Guide:

Oracle Web ADI brings Oracle E-Business Suite functionality to a spreadsheet where familiar data entry and modeling techniques can be used to complete Oracle E-Business Suite tasks. You can create formatted spreadsheets on your desktop that allow you to download, view, edit, and create Oracle E-Business Suite data that you can then upload. Use this guide to implement Oracle Web ADI and for information on defining mappings, layouts, style sheets, and other setup options.

Oracle Workflow Administrator's Guide:

This guide explains how to complete the setup steps necessary for any product that includes workflow-enabled processes. It also describes how to manage workflow processes and business events using Oracle Applications Manager, how to monitor the progress of runtime workflow processes, and how to administer notifications sent to workflow users.

Oracle Workflow API Reference:

This guide describes the APIs provided for developers and administrators to access Oracle Workflow.

Oracle Workflow Developer's Guide:

This guide explains how to define new workflow business processes and customize existing Oracle Applications-embedded workflow processes. It also describes how to define and customize business events and event subscriptions.

Oracle Workflow User's Guide:

This guide describes how users can view and respond to workflow notifications and monitor the progress of their workflow processes.

Oracle Approvals Management Implementation Guide:

Use Oracle Approvals Management (AME) to define the approval rules that determine the approval processes for Oracle applications.

Oracle Business Intelligence Discoverer Administration Guide:

Use this guide to find out how to set up and maintain a Discoverer system after installation. It covers how to use Discoverer Administrator to: create and maintain End User Layers; to set up business areas, folders and items; to help users find information by defining joins, calculated items, and conditions; and to improve Discoverer performance.

Oracle Business Intelligence Discoverer Plus User's Guide:

Use this guide to find out how to retrieve and analyze data by creating worksheets and charts, and how to publish those results. It covers the most common tasks you will perform with Discoverer Plus (for example, drilling and pivoting), along with reference information and useful examples. It includes an appendix containing detailed calculation examples.

Oracle Business Intelligence Discoverer Viewer User's Guide:

Use this guide to find out how to analyze data in worksheets that have already been created in Discoverer Plus. It covers the most common tasks you will perform with Discoverer Viewer (for example, drilling and pivoting), along with reference information and useful examples.

Oracle Embedded Data Warehouse Implementation Guide:

This guide describes how to implement Embedded Data Warehouse, including how to set up the intelligence areas.

Oracle Embedded Data Warehouse Install Guide:

This guide describes how to install Embedded Data Warehouse, including how to create database links and create the end user layer (EUL).

Oracle Embedded Data Warehouse User Guide:

This guide describes how to use Embedded Data Warehouse reports and workbooks to

analyze performance.

Oracle Enterprise Performance Foundation User's Guide:

This guide describes Oracle Enterprise Performance Foundation, an open and shared repository of data and business rules that provides the framework for all of the applications in the Corporate Performance Management set of products. It describes the product features that allow you to manage repository metadata and enable you to generate management reports and perform analyses.

Oracle Financial Services Implementation Guide:

This guide provides information about setting up Oracle Financial Services (OFS) applications in Release 12.

Oracle Financial Services Reference Guide:

This guide provides reference material for Oracle Financial Services applications in Release 12, such as Oracle Transfer Pricing, and includes technical details about application use as well as general concepts, equations, and calculations.

Oracle Financial Services Reporting Administration Guide:

This guide describes the reporting architecture of Oracle Financial Services applications in Release 12, and provides information on how to view these reports.

Oracle General Ledger Implementation Guide:

This guide provides information on how to implement Oracle General Ledger. Use this guide to understand the implementation steps required for application use, including how to set up Accounting Flexfields, Accounts, and Calendars.

Oracle General Ledger Reference Guide:

This guide provides detailed information about setting up General Ledger Profile Options and Applications Desktop Integrator (ADI) Profile Options.

Oracle General Ledger User's Guide:

This guide provides information on how to use Oracle General Ledger. Use this guide to learn how to create and maintain ledgers, ledger currencies, budgets, and journal entries. This guide also includes information about running financial reports.

Oracle Profitability Manager User's Guide:

This guide describes Profitability Manager, which provides a rich set of features that support complex models to analyze your business. These features include a powerful allocation engine that supports many allocation methodologies, Activity-Based Management calculations that provide activity costs, rolled up costs and statistics, activity rates, and cost object unit costs, and customer profitability calculations to consolidate customer accounts, aggregate customer data, and determine profitability results.

Integration Repository

The Oracle Integration Repository is a compilation of information about the service endpoints exposed by the Oracle E-Business Suite of applications. It provides a complete catalog of Oracle E-Business Suite's business service interfaces. The tool lets users easily discover and deploy the appropriate business service interface for integration with any system, application, or business partner.

The Oracle Integration Repository is shipped as part of the E-Business Suite. As your instance is patched, the repository is automatically updated with content appropriate for the precise revisions of interfaces in your environment.

Do Not Use Database Tools to Modify Oracle Applications Data

Oracle **STRONGLY RECOMMENDS** that you never use SQL*Plus, Oracle Data Browser, database triggers, or any other tool to modify Oracle Applications data unless otherwise instructed.

Oracle provides powerful tools you can use to create, store, change, retrieve, and maintain information in an Oracle database. But if you use Oracle tools such as SQL*Plus to modify Oracle Applications data, you risk destroying the integrity of your data and you lose the ability to audit changes to your data.

Because Oracle Applications tables are interrelated, any change you make using an Oracle Applications form can update many tables at once. But when you modify Oracle Applications data using anything other than Oracle Applications, you may change a row in one table without making corresponding changes in related tables. If your tables get out of synchronization with each other, you risk retrieving erroneous information and you risk unpredictable results throughout Oracle Applications.

When you use Oracle Applications to modify your data, Oracle Applications automatically checks that your changes are valid. Oracle Applications also keeps track of who changes information. If you enter information into database tables using database tools, you may store invalid information. You also lose the ability to track who has changed your information because SQL*Plus and other database tools do not keep a record of changes.

Introduction to Oracle Transfer Pricing

This chapter provides an introduction to Oracle Transfer Pricing and discusses its place in the Oracle Financial Services (OFS) group of applications.

This chapter covers the following topics:

- Overview of Oracle Transfer Pricing
- Oracle Transfer Pricing and Other Oracle Financial Services Applications

Overview of Oracle Transfer Pricing

Oracle Transfer Pricing (FTP) is the industry-standard software application for implementing a matched rate transfer pricing mechanism. Recognizing the value of matched rate transfer pricing, financial institutions are increasingly incorporating it in their performance measurement systems.

Matched rate transfer pricing overcomes the shortcomings of traditional transfer pricing approaches, such as unprofitable growth, repricing risk, and rate risk trap, by using multiple transfer rates instead of the single rate that traditional approaches advocate. Under the multiple rate approach, assets and liabilities are given transfer rates that reflect their specific maturity and repricing characteristics. See: *The Transfer Pricing Concept, Oracle Financial Services Reference Guide*.

Oracle Transfer Pricing calculates transfer rates at the lowest possible level of detail in your institution's balance sheet, the transaction record level. You can generate accurate charges and credits for all sources and uses of funds for your institution and measure its interest profitability at the transaction record level.

Oracle Transfer Pricing Key Benefits

To ensure accurate results, Oracle Transfer Pricing combines advanced methodologies with a flexible and easy-to-interpret reporting approach. Oracle Transfer Pricing allows you to:

- Determine the account level spread earned on assets and liabilities, and the spread

earned or lost as a result of interest rate risk exposure.

- Quantify and manage the implicit rate bet that results from your balance sheet management practices.
- Hold business units accountable for what they can control: pricing and profitability.
- Use account-level match funded spreads to produce account, customer, product, and business unit performance measures.

Related Topics

Oracle Transfer Pricing and Other Oracle Financial Services Applications, page 1-2

Overview of the Oracle Transfer Pricing Process, page 2-1

Oracle Transfer Pricing and Other Oracle Financial Services Applications

Oracle Transfer Pricing (FTP) is based on Oracle Enterprise Performance Foundation (EPF). EPF is the central, integrated data source on which Oracle Financial Services (OFS) applications in Release 12 are built. OFS applications form a comprehensive decision support solution that significantly enhances transfer pricing, budgeting and planning, asset/liability management, and performance measurement functions across a financial institution. In addition to Oracle Transfer Pricing, the OFS group of applications includes the following applications:

Oracle Budgeting and Planning

Budgeting and Planning provides performance-based planning. It integrates cash flow balance sheet and net income forecasting capabilities with the scalability and customizable framework of Oracle Financial Analyzer, part of the Oracle Express group of data access and analysis tools.

Oracle Enterprise Performance Foundation

Oracle Enterprise Performance Foundation is a standalone data model with prepackaged data elements that provides specific functionality for the financial services industry. EPF is also the foundation for the entire suite of OFS applications, providing the database structures necessary to support the individual business applications.

Oracle Risk Manager

Oracle Risk Manager allows you to analyze interest rate risk, and forecast cash flows, interest income, and market value to manage rate risk. Oracle Risk Manager provides both deterministic and stochastic methods for performing valuations and income simulations.

Oracle Profitability Manager

Oracle Profitability Manager provides a comprehensive and flexible cost and equity allocation framework. It allows you to develop processes for measuring profitability against any number of dimensions including account, customer, product, segment, and organization.

Related Topics

Overview of Oracle Transfer Pricing, page 1-1

Oracle Transfer Pricing Integrations, page 1-3

Oracle Transfer Pricing Integrations

Oracle Transfer Pricing integrates with the following modules:

- Oracle Profitability Manager
- Oracle Budgeting and Planning
- Oracle Risk Manager

A transfer-priced balance sheet is merely the beginning. You can combine Oracle Transfer Pricing results with noninterest income and expense information populated at the account level with Oracle Profitability Manager to measure total profitability based on a user-definable combination of dimensions. You can also access results in Oracle Budgeting and Planning or Oracle Risk Manager.

Related Topics

Oracle Transfer Pricing and Other Oracle Financial Services Applications, page 1-2

Overview of Oracle Transfer Pricing, page 1-1

The Oracle Transfer Pricing Process

This chapter describes the set of steps that you need to follow to transfer price your balance sheet using Oracle Transfer Pricing.

This chapter covers the following topics:

- Overview of the Oracle Transfer Pricing Process
- Setting Transfer Pricing Rules
- Setting Prepayment Rules
- Setting and Executing the Transfer Pricing Process Rule
- Defining Payment Patterns
- Defining Repricing Patterns
- Performing Cash Flow Edits
- Creating Interest Rate Codes
- Accessing Transfer Pricing Detail Cash Flow Results for Audit Purposes
- Analyzing Results
- Reviewing Processing Errors
- Reprocessing Erroneous Accounts
- Reconciling the Data

Overview of the Oracle Transfer Pricing Process

Oracle Transfer Pricing (FTP) is based on the Enterprise Performance Foundation (EPF). EPF is the central, integrated data source on which Oracle Financial Services (OFS) applications in Release 12 are built. This description of the Oracle Transfer Pricing process assumes that your system administrator has set up the EPF data repository and has populated it with your enterprise-wide business data.

Important: Oracle Transfer Pricing allows you to transfer price instruments, such as mortgages and commercial loans, stored in the EPF Account tables, also known as Instrument tables, as well as aggregated information, such as, cash and other assets, and equity, residing in the Management Ledger table, also known as FEM_BALANCES.

Consequently, while transfer pricing you need to select the Account tables as the data source for instruments and the Management Ledger Table for aggregated information. You may also choose to migrate your aggregated balances from the Management Ledger table to custom Account tables, effectively treating these aggregated balances as instrument records for processing purposes.

The Oracle Transfer Pricing process comprises the following steps:

Important: Although the following list of steps is sequential, not all the users need to follow all the steps. While some of these steps might not be applicable to your product portfolio, some others are optional, and it is up to you to decide whether you want to include them to fine tune your transfer pricing results. All the required steps are explicitly marked as mandatory in the list of steps, as well as in the sections where they are described in detail.

1. Reconciling the data, page 2-59
2. Cleansing the data by Performing Cash Flow Edits, page 2-52
3. Capturing instrument behavior by
 - Defining Payment Patterns, page 2-44
 - Defining Repricing Patterns, page 2-48
4. (Mandatory) Deciding on historical rate information and managing it by Creating Interest Rate Codes, page 2-53
5. (Mandatory) Setting Transfer Pricing Rules, page 2-27
6. Setting Prepayment Rules, page 2-27
7. (Mandatory) Setting and Executing the Transfer Pricing Process Rule, page 2-36
8. Reviewing processing errors, page 2-58
9. Accessing Transfer Pricing Detail Cash Flow Results for Audit Purposes, page 2-55

10. Analyzing results, page 2-57
11. Reprocessing erroneous accounts, page 2-59

Setting Transfer Pricing Rules

Setting Transfer Pricing rules is one of the mandatory steps in the Oracle Transfer Pricing process. Unless you set Transfer Pricing rules, you cannot transfer price your products. A Transfer Pricing rule is used to manage the association of transfer pricing methodologies to various product-currency combinations. It can also be used to manage certain parameters used in option costing. See: Transfer Pricing Methodologies and Rules, page 2-4.

Setting the Transfer Pricing rules is a two-step process. You need to first create rules and then versions. Transfer pricing methodologies are associated to various product-currency combinations within a version. A Transfer Pricing rule can have more than one version. See: Transfer Pricing Rules, page 12-1.

To reduce the amount of effort required to define the transfer pricing methodologies for various products and currencies, Oracle Transfer Pricing allows you to define transfer pricing methodologies using node level and conditional assumptions.

Node Level Assumptions

Oracle Transfer Pricing uses the Line Item Dimension to represent a financial institution's product portfolio. Using this dimension, you can organize your product portfolio in a hierarchical structure and define parent-child relationships among different nodes of your product hierarchies. This significantly reduces the amount of work required to define transfer pricing and prepayment methodologies.

You can define transfer pricing and prepayment methodologies at any level of your product hierarchy. Children of parent nodes on a hierarchy automatically inherit the methodologies defined for the parent nodes. However, methodologies directly defined for a child take precedence over those at the parent level. See: Defining Transfer Pricing Methodologies Using Node Level Assumptions, page 2-19.

Conditional Assumptions

This feature allows you to segregate your product portfolio based on common characteristics, such as term to maturity, origination date, and repricing frequency, and assign specific transfer pricing methodologies to each of the groupings.

For example, you can slice a portfolio of commercial loans based on repricing characteristics and assign one global set of transfer pricing and prepayment methods to the fixed-rate loans and another to the floating-rate loans. See: Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Defining Transfer Pricing Methodologies, page 12-3

Transfer Pricing Methodologies and Rules

The transfer pricing methodologies supported by Oracle Transfer Pricing can be grouped into the following two categories:

- **Cash Flow Transfer Pricing Methods:** Cash flow transfer pricing methods are used to transfer price instruments that amortize over time. They generate transfer rates based on the cash flow characteristics of the instruments.

In order to generate cash flows, the system requires a detailed set of transaction-level data attributes, such as, origination date, outstanding balance, contracted rate, and maturity date, which resides only in the Account tables. Consequently, cash flow methods apply only if the data source is Account tables. Data stored in the Management Ledger Tables reflects only accounting entry positions at a particular point in time and do not have the required financial details to generate cash flows, thus preventing you from applying cash flows methodologies to this data.

The cash flow methods are also unique in that Prepayment rules are used only with these methods. You can select the required Prepayment rule when defining a Transfer Pricing Process rule.

See: Setting Prepayment Rules, page 2-27 and Cash Flow Calculations, *Oracle Financial Services Reference Guide*.

Oracle Transfer Pricing supports the following cash flow transfer pricing methods:

- Duration, page 2-5
 - Weighted Average Cash Flow, page 2-6
 - Zero Coupon Pricing, page 2-7
- **Noncash Flow Transfer Pricing Methods:** These methods do not require the calculation of cash flows. While some of the noncash flow methods are available only with the Account tables data source, some are available with both the Account and Ledger table data sources.

Oracle Transfer Pricing supports the following noncash flow transfer pricing methods:

- Moving Averages, page 2-9
- Straight Term, page 2-10

- Spread from Interest Rate Code, page 2-11
- Spread from Note Rate, page 2-11
- Redemption Curve, page 2-12
- Unpriced Account, page 2-13

Oracle Transfer Pricing also allows Mid-period Repricing, page 2-14. This option allows you to take in to account the impact of high market rate volatility while generating transfer prices for your products. However, the mid-period repricing option applies only to adjustable rate instruments and is available only for certain noncash flow transfer pricing methods.

Related Topics

Setting Transfer Pricing Rules, page 2-3

Defining Transfer Pricing Methodologies, page 12-3

Duration

The Duration method uses the MacCauley duration formula:

$$\text{Duration} = \frac{\sum_{n=1}^N \left[\frac{CF_n}{(1+r)^m} \times t_n \right]}{\sum_{n=1}^N \left[\frac{CF_n}{(1+r)^m} \right]}$$

In this formula:

N = Total number of payments from Start Date until the earlier of repricing or maturity

CF_n = Cash flow (such as regular principal, prepayments, and interest) in period n

r = Periodic coupon rate on instrument (current rate/payments per year)

m = Remaining term to cash flow/active payment frequency

tn = Remaining term to cash flow n , expressed in years

Oracle Transfer Pricing derives the MacCauley duration based on the cash flows of an instrument as determined by the characteristics specified in the Account Table and using your specified prepayment rate, if applicable. The duration formula calculates a single term, that is, a point on the yield curve used to transfer price the instrument is being analyzed.

- Current rate is defined as current net rate if the processing option "Model with Gross Rates" is not selected and current gross rate if the option is selected. The current rate is used as the discount rate for each cash flow.
- Remaining term to cash flow is the difference between the date of each cash flow and the modeling start date for that instrument.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Weighted Average Cash Flow

The Weighted Average Cash Flow (WACF) method builds on the theoretical concepts of duration. As shown earlier, duration calculates a weighted-average term by weighting each time period, n , with the present value of the cash flow (discounted by the rate on the instrument) in that period.

Since the goal of the WACF method is to calculate a weighted average transfer rate, it weights the transfer rate in each period, y_n , by the present value for the cash flow of that period. Furthermore, the transfer rates are weighted by an additional component, *time*, to account for the length of time over which a transfer rate is applicable. The time component accounts for the relative significance of each strip cash flow to the total transfer pricing interest income/expense. The total transfer pricing interest income/expense on any cash flow is a product of that cash flow, the transfer rate, and the term. Hence, longer term cash flows will have relatively larger impact on the average transfer rate. The WACF method can be summarized by the following formula:

$$\text{Weighted-Average} = \bar{y} = \frac{\sum_{n=1}^N \left[y_n \times \frac{CF_n}{(1+r)^m} \times t_n \right]}{\sum_{n=1}^N \left[\frac{CF_n}{(1+r)^m} \times t_n \right]}$$

In this formula:

N = Total number of payments from Start Date until the earlier of repricing or maturity

CF_n = Cash flow (such as regular principal, prepayments, and interest) in period n

r = Periodic coupon rate on instrument (current rate/payments per year)

m = Remaining term to cash flow n /active payment frequency

tn = Remaining term to cash flow n , expressed in years

y_n = Transfer rate in period n

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Zero Coupon Pricing

The Zero Coupon Pricing (ZCP) method takes into account common market practices in valuing fixed rate amortizing instruments. For example, all Treasury strips are quoted as discount factors. A discount factor represents the amount paid today to receive \$1 at maturity date with no intervening cash flows (that is, zero coupon).

The Treasury discount factor for any maturity (as well as all other rates quoted in the market) is always a function of the discount factors with shorter maturities. This ensures that no risk-free arbitrage exists in the market. Based on this concept, one can conclude that the rate quoted for fixed rate amortizing instruments is also a combination of some set of market discount factors. Discounting the monthly cash flows for that instrument (calculated based on the constant instrument rate) by the market discount factors generates the par value of that instrument (otherwise there is arbitrage).

ZCP starts with the assertion that an institution tries to find a funding source that has the same principal repayment factor as the instrument being funded. In essence, the institution strip funds each principal flow using its funding curve (that is, the transfer pricing yield curve). The difference between the interest flows from the instrument and its funding source is the net income from that instrument.

Next, ZCP tries to ensure consistency between the original balance of the instrument and the amount of funding required at origination. Based on the transfer pricing yield used to fund the instrument, the ZCP solves for a single transfer rate that would amortize the funding in two ways:

- Its principal flows match those of the instrument.
- The Present Value (PV) of the funding cash flows (that is, the original balance) matches the original balance of the instrument.

ZCP uses zero coupon factors (derived from the original transfer rates, see the example below) because they are the appropriate vehicles in strip funding (that is, there are no intermediate cash flows between origination date and the date the particular cash flow is received). The zero coupon yield curve can be universally applied to all kinds of instruments.

This approach yields the following formula to solve for a weighted average transfer rate based on the payment dates derived from the instrument's payment data.

Zero Coupon Pricing = $y =$

$$100x \left[\frac{B_0 - \sum_{n=1}^N (B_{n-1} \times DTP_n) + \sum_{n=1}^N (B_n \times DTP_n)}{N} - \sum_{n=1}^N (B_{n-1} \times DTP_n) \right] \times \frac{1}{p}$$

In this formula:

B_0 = Beginning balance at time, 0

B_{n-1} = Ending balance in previous period

B_n = Ending balance in current period

DTP_n = Discount factor in period n based on the TP yield curve

N = Total number of payments from Start Date until the earlier of repricing or maturity

p = Payments per year based on the payment frequency; (for example, monthly payments gives $p=12$)

Deriving Zero Coupon Discount Factors: An Example

This table illustrates how to derive zero coupon discount factors from monthly pay transfer pricing rates:

Term in Months	(a) Monthly Pay Transfer Rates	(b) Monthly Transfer Rate: (a)/12	(c) Numerator (Monthly Factor): 1+ (b)	(d) PV of Interest Payments: (b)*Sum((f)/10 0 to current row	(e) Denominator (1 - PV of Int Pmt): 1 - (d)	(f) Zero Coupon Factor: [(e)/(c) * 100
1	3.400%	0.283%	1.002833	0.000000	1.000000	99.7175
2	3.500%	0.292%	1.002917	0.002908	0.997092	99.4192
3	3.600%	0.300%	1.003000	0.005974	0.994026	99.1053

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Moving Averages

Under this method, a user definable moving average of any point on the transfer pricing yield curve can be applied to a transaction record to generate transfer prices. For example, you can use a 12-month moving average of the 12-month rate to transfer price a particular product.

The following options become available on the user interface (UI) with this method:

- **Interest Rate Code:** Select the Interest Rate Code to be used as the yield curve to generate transfer rates.
- **Yield Curve Term:** The Yield Curve Term defines the point on the Interest Rate Code that is used.
- **Historical Term:** The Historical Term defines the period over which the average is calculated.

The following table illustrates the difference between the yield curve and historical terms.

Yield and Historical Terms: An Example

Moving Average	Yield Curve Term	Historical Term
Six-month moving average of 1 year rate	1 year (or 12 months)	6 months
Three-month moving average of the 6 month rate	6 months	3 months

The range of dates is based on the effective date minus the historical term plus one, because the historical term includes the effective date. Oracle Transfer Pricing takes the values of the yield curve points that fall within that range and does a straight average on them.

For example, if Effective Date is Nov 21, the Yield Curve Term selected is Daily, and the Historical Term selected is 3 Days, then, the system will calculate the three-day moving average based on the rates for Nov 19, 20, and 21. The same logic applies to monthly or annual yield terms.

Note: The Moving Averages method applies to either data source: Management Ledger Table or Account Tables.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Straight Term

When you select the Straight Term method, the system derives the transfer rate using the last repricing date and the next repricing date for nonfixed rate instruments, and the origination date and the maturity date for fixed rate instruments.

1. Standard Calculation Mode:
 1. For Fixed Rate Products (Repricing Frequency = 0), use Yield Curve Date = Origination Date, Yield Curve Term = Maturity Date - Origination Date.
 2. For Adjustable Rate Products (Repricing Frequency > 0)
 - For loans still in tease period (tease end date > Effective Date, and tease end date > origination date), use Origination Date and Tease End Date - Origination Date.
 - For loans not in tease period, use Last Repricing Date and Repricing Frequency.
2. Remaining Term Calculation Mode:
 1. For Fixed Rate Products, use Effective Date and Maturity - Effective Date.
 2. For Adjustable Rate Products, use Effective Date and Next Repricing Date - Effective Date.

The following options become available on the application with this method:

- **Interest Rate Code:** Select the Interest Rate Code to be used for transfer pricing the account.
- **Mid-Period Repricing Option:** Select the check box beside this option to invoke the Mid-Period Repricing option.

Note: The Straight Term method applies only to accounts that use Account Tables as the data source.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Spread from Interest Rate Code

Under this method, the transfer rate is determined as a fixed spread from any point on an Interest Rate Code. The following options become available on the application with this method:

- **Interest Rate Code:** Select the Interest Rate Code for transfer pricing the account.
- **Yield Curve Term:** The Yield Curve Term defines the point on the Interest Rate Code that will be used to transfer price. If the Interest Rate Code is a single rate, the Yield Curve Term is irrelevant. Select Days, Months, or Years from the drop-down list, and enter the number.
- **Lag Term:** While using a yield curve from an earlier date than the Assignment Date, you need to assign the Lag Term to specify a length of time prior to the Assignment Date.
- **Rate Spread:** The transfer rate is a fixed spread from the rate on the transfer rate yield curve. The Rate Spread field allows you to specify this spread.
- **Assignment Date:** The Assignment Date allows you to choose the date for which the yield curve values are to be picked up. Choices available are the Calendar Period End Date, Last Repricing Date, or Origination Date.
- **Mid-Period Repricing Option:** Select the check box beside this option to invoke the Mid-Period Repricing option.

Note: The Spread From Interest Rate Code method applies to either data source: Management Ledger Table or Account Tables.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Spread from Note Rate

To generate transfer prices using this method, you need to provide just one parameter: a rate spread. This spread is added or subtracted from the coupon rate of the underlying transaction to generate the final transfer rate for that record.

While entering the rate spread, make sure to input it with the appropriate positive or negative sign, as illustrated in the following table. The first row describes a situation where you are transfer pricing an asset and want to have a positive matched spread for it (the difference between the contractual rate of the transaction and the transfer rate is positive). Here, you need to enter a negative rate spread.

Example of Rate Spread

Account Type	Matched Spread	Sign of Rate Spread
Asset	Positive (Profitable)	Negative
Asset	Negative (Unprofitable)	Positive
Liability or Equity	Positive (Profitable)	Positive
Liability or Equity	Negative (Unprofitable)	Negative

The following option becomes available on the application when you select this method:

- **Mid-Period Repricing Option:** Select the check box beside this option to invoke the Mid-Period Repricing option.

Note: The Spread From Note Rate method applies only to accounts that use Account Tables as their data source.

Spread from Note Rate Computations

The Spread from Note Rate method involves a calculation of interest rate from the specified Interest Rate Code and adding the given margin to this rate. Next, covenants, such as rate change minimum and lifetime caps and floors are applied, if any, in addition to rounding it off to appropriate number of decimal points.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

Redemption Curve

This method allows you to select multiple term points from your transfer pricing yield curve and calculate an average transfer rate based on the weights you assign to each term point. The following options become available on the application with this method:

- **Interest Rate Code:** Select the Interest Rate Code which you want to use as the transfer pricing yield curve.
- **Assignment Date:** The Assignment Date allows you to choose the date for which

the yield curve values will be picked up. Choices available are the Calendar Period End Date, Last Repricing Date, or Origination Date.

- **Percentages/Term Points:** See: Defining the Redemption Curve Methodology, page 12-9.
- **Mid-Period Repricing Option:** Select the check box beside this option to invoke the Mid-Period Repricing option.

Note: The Redemption Curve method applies to either data source: Management Ledger Table or Account Tables.

Related Topics

Defining Transfer Pricing Methodologies, page 12-3

Unpriced Account

Under the unpriced account method, the transfer rate for the account is defined as the weighted average of the Line Item dimension members (products). While using the unpriced account methodology, you can specify whether the weighted average of transfer rates has to be taken across all organizational units or for accounts only within that organizational unit.

The following options become available on the application with this method:

- **Add Dimension Values:** Allows you to select the Line Item dimension members (products) whose weighted average transfer rate will be assigned to the product being defined.

Caution: You should not base an unpriced account on another unpriced account, since the processing hierarchy does not properly allow for it.

- **Across all Organization Units:** Allows you to specify whether weighted average of transfer rates has to be taken across all organizational units. See: Defining the Unpriced Account Methodology, page 12-10.

Note: The Unpriced Account method applies only to accounts that use Management Ledger Table as their data source.

Related Topics

Defining Transfer Pricing Methodologies, page 12-3

Transfer Pricing Methods and the Mid-Period Repricing Option

The mid-period repricing option allows you to take into account the impact of high market rate volatility while generating transfer rates for your products. However, the mid-period repricing option applies only to adjustable rate instruments and is available only for the following noncash flow transfer pricing methods:

- Straight Term.
- Spread from Interest Rate Code.
- Spread from Note Rate.
- Redemption Curve.

Note: For the Spread from Interest Rate Code and Redemption Curve methods, the Assignment Date must be set to Last Repricing Date in order to choose Mid-Period Repricing.

The rationale behind mid-period repricing is as follows. If you do not select the Mid-Period Repricing option, Oracle Transfer Pricing computes the transfer rate for an adjustable rate instrument based upon its last repricing date. The assumption behind this method of calculation is that the input transfer rate for a month should be the daily average transfer rate for that entire month. Consequently, all instruments repricing in that month derive their transfer rates from the same (average) transfer pricing yield curve. However, this approach misstates the transfer rate, in periods when the interest rate level has moved substantially since the last repricing.

Take the example of a one-year adjustable rate loan. Suppose it reprices on the 15th of the month, and that transfer rates have moved up 200 basis points since the last reprice. In such a case, the theoretically pure transfer rate for the first half of the month should be 200 basis points lower than the transfer rate for the second half of the month. In order to apply such theoretical accuracy to your transfer pricing results, you should select the Mid-Period Repricing option.

Mid-Period Repricing Computations

The Mid-Period Repricing option uses two columns in the Account Tables (Current- and Prior- Repricing Period Average Daily Balance: CUR_TP_PER_ADB, PRIOR_TP_PER_ADB) that are exclusively devoted to this option. These columns must be accurately populated for the Mid-Period Repricing results to be accurate.

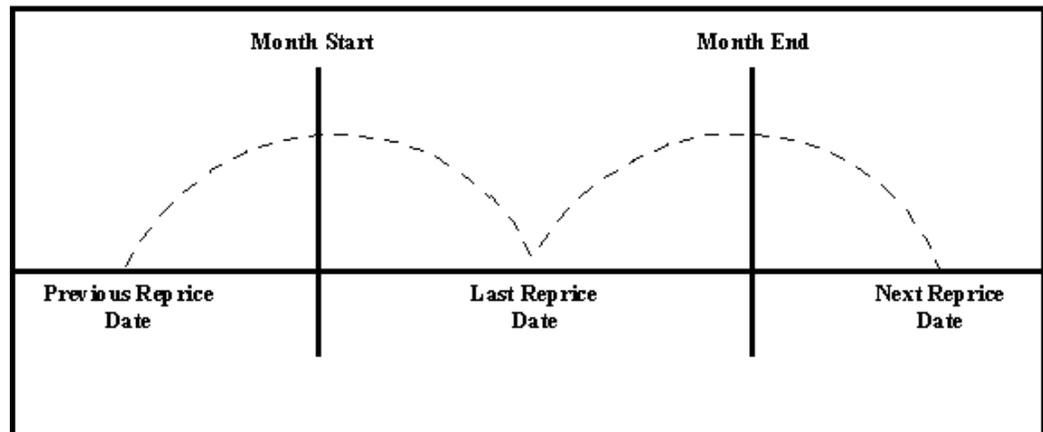
The Mid-Period Repricing Computation process comprises the following steps:

1. Computation of transfer rate for current repricing period.
2. If the computed last repricing date > beginning of processing month, roll back to prior repricing date.

3. Computation of prior period transfer rate.
4. Repetition of steps 2 and 3 as necessary.
5. Computation of the final transfer rate by weighting the results (from current and previous repricing periods) by average balances and days.
6. Application of the final transfer rate to the instrument record.

Typical Calculations

The following diagram depicts a typical Mid-Period Repricing situation where the instrument reprices during the current processing month.



If an instrument reprices during the current processing month, then there are multiple repricing periods spanning the current month. In this example, there are two repricing periods in the current processing month and the computed last repricing date > beginning of processing month. Consequently, the repricing dates need to be rolled back by the repricing frequency until the Prior Last Repricing Date (Prior LRD) <= Beginning of Month and the Mid-Period Repricing Computation process should be executed as follows:

1. Computation of transfer rate for current repricing period.
 - Transfer Pricing Term: Next Reprice Date - Last Reprice Date
 - Transfer Pricing Date: Last Reprice Date
 - Number of Days at that Rate: End of Month + 1- Last Reprice Date

Note: If the Computed Next Reprice Date (the next repricing date for a given repricing period) is less than or equal to the End of Month, then the Number of Days calculation uses the

Computed Next Reprice Date in place of End of Month. In other words, Number of Days equals the Minimum (End of Month + 1, Computed Next Reprice Date) - Maximum (Beginning of Month, Computed Last Reprice Date).

This example assumes the use of the Straight Term transfer pricing method. The following table describes the logic for the computation of the transfer rates for each method.

Method	Date for Rate Lookup	Terms	Interest Rate Code	Spread
Straight Term	Beginning of Reprice Period	Transfer Pricing Term	Specified in Transfer Pricing rule	Not Applicable
Spread from Interest Rate Code	Beginning of Reprice Period (adjust by Lag Term in TP Rule)	Specified in Transfer Pricing rule	Specified in Transfer Pricing rule	Specified in Transfer Pricing rule
Spread from Note Rate	Beginning of Reprice Period	Transfer Pricing Term	Interest Rate Code from Record	Specified in Transfer Pricing rule
Redemption Curve	Beginning of Reprice Period	Specified in Transfer Pricing rule	Specified in Transfer Pricing rule	Not Applicable

2. If the computed last repricing date > beginning of processing month, roll back to prior repricing date.

Since the Last Repricing Date is greater than the Beginning of the Processing month, the Roll Back is done as follows:

Computed Next Reprice Date is reset to Last Reprice Date

Computed Last Repricing Date is reset to Last Repricing Date - Reprice Frequency (Prior LRD)

3. Computation of prior period transfer rate.
 - Transfer Pricing Term: Last Reprice Date - Prior LRD

- Transfer Pricing Date: Prior LRD
- Number of Days at that Rate: Last Reprice Date - Beginning of Month

Note: If the Computed Last Reprice Date (the last repricing date for a given repricing period) is greater than the Beginning of Month, then the Number of Days calculation uses Computed Last Reprice Date in place of the Beginning of Month. In other words, Number of Days equals Minimum (End of Month + 1, Computed Next Reprice Date) - Maximum (Beginning of Month, Computed Last Reprice Date).

4. Repetition of steps 2 and 3 as necessary.

In this example, only one iteration is needed because Prior LRD is less than the Beginning of the Month.

5. Computation of the final transfer rate by weighting the results (from current and previous repricing periods) by average balances and days.

$$\frac{((\text{CUR_TP_PER_ADB} * \text{Cur Period Transfer Rate} * \text{Cur Period Days}) + \sum (\text{PRIOR_TP_PER_ADB} * \text{Prior Period Transfer Rate} * \text{Prior Period Days}))}{((\text{CUR_TP_PER_ADB} * \text{Cur Period Days}) + \sum (\text{PRIOR_TP_PER_ADB} * \text{Prior Period Days}))}$$

The calculation makes the following assumptions:

- CUR_TP_PER_ADB is the balance applying since the last reprice date
 - PRIOR_TP_PER_ADB is the balance applying to all prior repricing periods
6. Application of the final transfer rate to the instrument record.

Exceptions to Typical Calculations

There are two exceptions to typical mid-period repricing computations:

Teased Loan Exception

When the TEASER_END_DATE is the first repricing date, it overrides all other values for LAST_REPRICE_DATE and NEXT_REPRICE_DATE. During the Teased Period, then, the Computed Last Repricing Date equals the Origination Date and the Computed Next Reprice Date equals the TEASER_END_DATE. Consequently:

- If the TEASER_END_DATE is greater than the AS_OF_DATE, the Mid-Period Repricing Does not apply. The logic to compute the Transfer rate is based upon term equal to the TEASER_END_DATE - ORIGINATION_DATE, date equals the ORIGINATION_DATE.

- When rolling backwards by repricing frequency, if the TEASER_END_DATE is greater than the Computed Last Repricing Date, Transfer Pricing computes the transfer rates for that period based on the teased loan exception.

Origination Date Exception

While performing mid-period repricing computations, Oracle Transfer Pricing assumes that if the origination date occurs during the processing month, the calculation of the number of days (used for weighting) originates on the first day of the month. This is a safe assumption because the PRIOR_TP_PER_ADB value shows this instrument was not on the books for the entire month. This impact is measured because the PRIOR_TP_PER_ADB value is used in computing the weighted average transfer rate. If Oracle Transfer Pricing were to shorten the number of days (as in the weighted average calculation), it would double-count the impact.

Origination Date Exception: An Example

The following table displays a situation where the origination date occurs during the processing month:

Period 1	Period 2	Period 3
Nov 1 - Nov 10	Nov 11 - Nov 20	Nov 21 - Nov 30
	Loan is originated	Loan reprices
Loan Balance = 0	Loan Balance = 100	Loan Balance = 100
Transfer Rate = 0	Transfer Rate = 6%	Transfer Rate = 8%
Days = 10	Days = 10	Days = 10
Weighting Balance = 50 = PRIOR_TP_PER_ADB	Weighting Balance = 50 = PRIOR_TP_PER_ADB	Weighting Balance = 100 = CUR_TP_PER_ADB

Note: The cumulative average daily balance for period 1 plus period 2 is 50.

Taking the origination date exception in to account, the Mid-Period Repricing calculation is done as follows:

$$(6\% * \$50 * 20 \text{ days}) + (8\% * \$100 * 10 \text{ days}) / (\$50 * 20 \text{ days} + \$100 * 10 \text{ days}) = 7\%$$

If period 1 was not taken into account, the result would have been, $(6\% * \$50 * 10 \text{ days}) + (8\% * \$100 * 10 \text{ days}) / (\$50 * 10 \text{ days} + \$100 * 10 \text{ days}) = 7.33\%$, which is incorrect.

Related Topics

Transfer Pricing Methodologies and Rules, page 2-4

Defining Transfer Pricing Methodologies, page 12-3

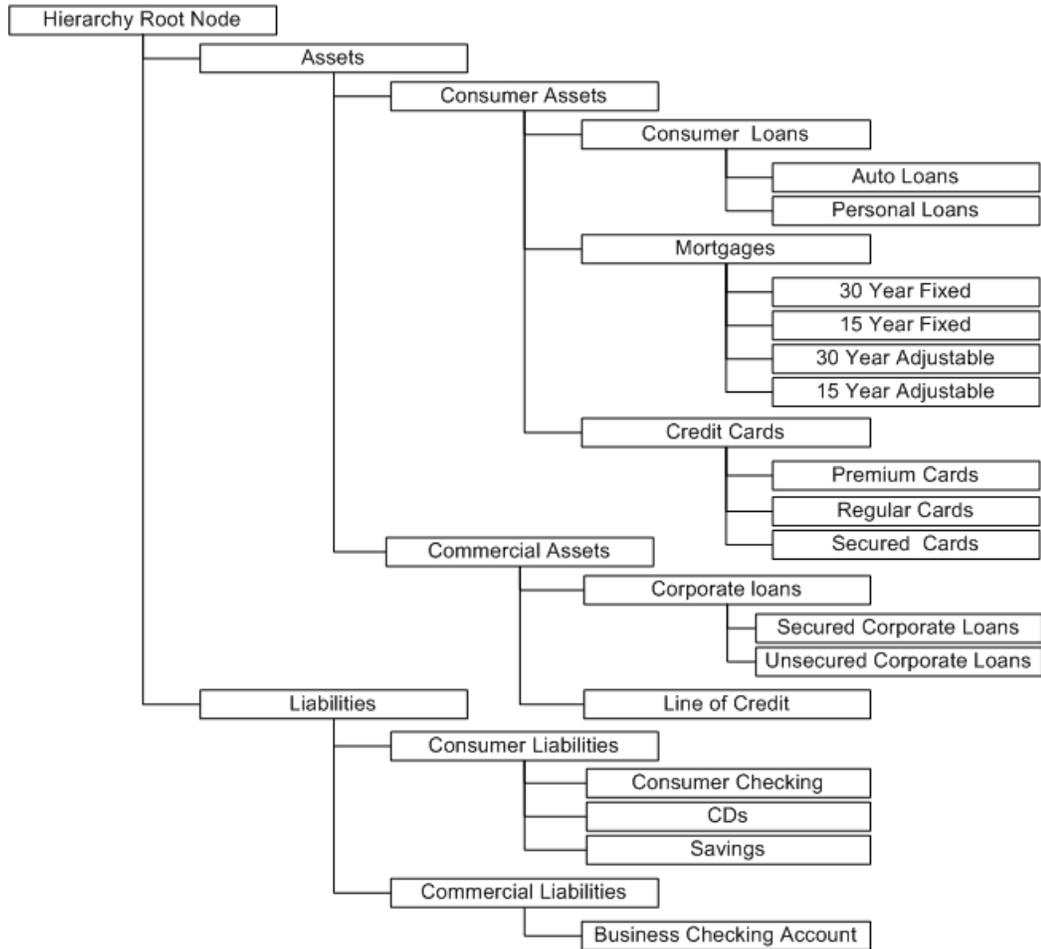
Defining Transfer Pricing Methodologies Using Node Level Assumptions

In Oracle Transfer Pricing, your product portfolio is represented using the Line Item Dimension. Node Level Assumptions allow you to define transfer pricing and prepayment assumptions at any level of the Line Item Dimension. The Line Item Dimension supports a hierarchical representation of your chart of account. So you can take advantage of the parent-child relationships among different nodes of your product hierarchies while defining transfer pricing and prepayment assumptions. Child nodes for which no assumption has been specified automatically inherit the methodology of their closest parent node.

Node level assumptions can be applied only to the product dimension and not to the currency dimension. When multiple rules are combined to form a processing rule, all rules must follow the same hierarchy and, thereby, the same product dimension. This simplifies the process of applying rules in the user interface. It is also not required for all rules to assign assumptions to the same nodes. Users may assign assumptions at different levels. If assumptions are specified using multiple dimensions, only one dimension will allow the use of node level assumptions.

Behavior of Node Level Assumptions

The following graphic displays a sample product hierarchy:



Suppose you want to transfer price this product hierarchy using the Spread from Interest Rate Code transfer pricing method except for the following products:

- Mortgages: You want to transfer price these using the Zero Coupon Cash Flow method.
- Credit Cards: You want to transfer price all but secured credit cards using the Spread from Note Rate method.

To transfer price in this manner, you need to attach transfer pricing methods to the nodes of the product hierarchy as follows:

- Hierarchy Root Node: Spread from Interest Rate Code
- Mortgages: Zero Coupon Cash Flow
- Credit Cards: Spread from Note Rate
- Secured Credit Cards: Spread from Interest Rate Code

The transfer pricing method for a particular product is determined by searching up the nodes in the hierarchy. Consider the Secured Credit Cards in the above example. Since Spread from IRC is specified at the hierarchy level, the system does not need to search any further to calculate the transfer rates for the Secured Credit Cards. However, for a Premium Credit Card the system searches up the hierarchical nodes for the first node that specifies a method. The first node that specifies a method for the Premium Credit Card is the Credit Card node and it is associated with the Spread from Note Rate method.

All parameters that are attached to a particular methodology (such as Interest Rate Code) are specified at the same level as the method. If multiple Interest Rate Codes are to be used, depending on the type of the product, the method would need to be specified at a lower level. For instance, if you want to use IRC 211178 for Consumer Products and IRC 3114 for Commercial Products, then the transfer pricing methodologies for these two products need to be specified at the Commercial Products and Consumer Products nodes.

You need not specify prepayment assumptions at the same nodes as transfer pricing methods. For example, each mortgage category can have a different prepayment method while the entire Mortgage node uses the Zero Coupon Cash Flow method for transfer pricing.

Related Topics

Setting Transfer Pricing Rules, page 2-3

Defining Transfer Pricing Methodologies, page 12-3

Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21

Conditional Assumptions, page 11-1

Associating Conditional Assumptions with Transfer Pricing Rules

Oracle Transfer Pricing enhances the setup and maintenance of methodologies by providing conditional logic (optional) to assign Transfer Pricing and Prepayment Methods to combinations of products and currencies.

You can now define Transfer Pricing and Prepayment methodologies using "IF-THEN-ELSE" logic based on the underlying characteristics of financial instruments, such as dates, rates, balances, and code values.

In addition, Conditional Assumptions can now be attached to any level of the Line Item Dimension hierarchy, allowing assumptions to be inherited from parent nodes to children nodes.

Oracle Transfer Pricing provides a set of user interfaces specially designed to easily build Conditional Assumptions. The logic included in a Conditional Assumption drives the specific Transfer Pricing or Prepayment method that the system would assign to a product-currency combination at run time.

For example, you can use the maturity date column on the commercial loans table to drive the assignment of Transfer Pricing Methods for all the records in that table. You can create one Conditional Assumption to convey the entire Transfer Pricing Methodology logic and attach it to the top-level node of the Line Item Dimension hierarchy representing the commercial loan portfolio. All nodes below the top-level node will inherit the same Transfer Pricing assumptions. See: Overview of Conditional Assumptions, page 11-1.

A Conditional Assumption is made of at least three (IF, THEN, and ELSE) clauses. Each clause is displayed in the user interface as a row. The following table displays a basic Conditional Assumption with three clauses:

Conditional Assumption with Three Clauses

Condition Term	Product Attribute	Operator	Value	TP Method
IF	Repricing Frequency	<>	0	
THEN				Cash Flow Weighted Term
ELSE				Straight Term

In the above table, the Cash Flow Weighted Term transfer pricing method is assigned to the designated products that have a repricing frequency not equal to zero. If the repricing frequency is zero, then the Straight Term Transfer Pricing Method is assigned.

Conditional Assumptions can be applied only on detailed accounts (data stored in the Account Tables) and reference only the Cash Flow and Dimension columns that are part of the Account Tables.

Related Topics

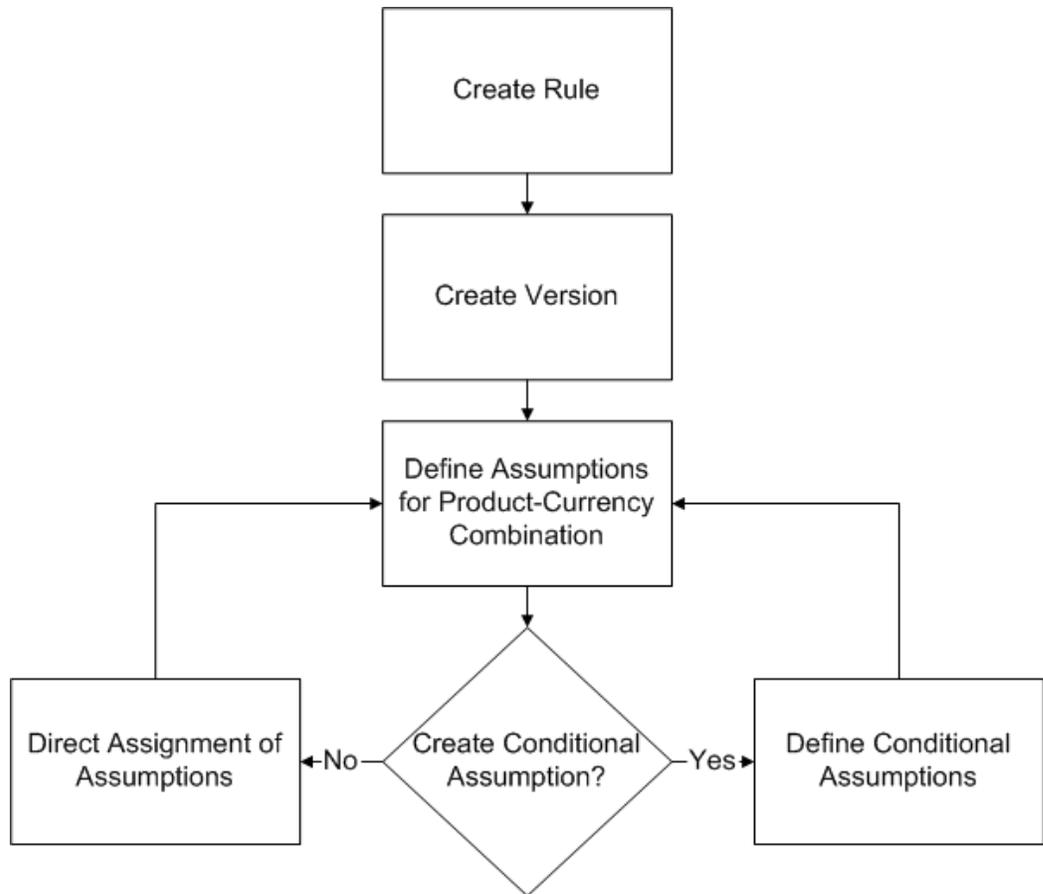
Setting Transfer Pricing Rules, page 2-3

Defining Transfer Pricing Methodologies, page 12-3

Availability of Conditional Assumptions

Conditional Assumptions cannot exist independently; they are an extension of Transfer Pricing and Prepayment Rules.

To define a Conditional Assumption, you need to complete a series of steps beforehand. The following diagram displays, at a high level, how the Conditional Assumptions functionality fits with the overall Create process of a Transfer Pricing rule.



Assigning Conditional Assumptions is a sub-process within the Create or Update Flow of the Transfer Pricing and Prepayment Rules. Once you create a Rule and a Version, you have two options for defining your transfer pricing methodologies for a product-currency combination.

- Direct assignment of a Transfer Pricing or a Prepayment Method to a product-currency combination.

This is the conventional method. See: Defining Transfer Pricing Methodologies, page 12-3 and Defining Prepayment Methodologies, page 13-3.

- Assignment of the methodology through a Conditional Assumption.

In this scenario, you would define IF-THEN-ELSE logic that will determine the Transfer Pricing or Prepayment Methodology for product-currency combinations. See: Conditional Assumptions, page 11-1.

Related Topics

Setting Transfer Pricing Rules, page 2-3

Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21

Structure of Conditional Assumptions

A Conditional Assumption comprises *blocks of logic*. These blocks are in turn built of *clauses* that exist within each block of logic.

There are three kinds of blocks:

- IF
- ELSE IF
- ELSE

The IF Block

The IF block is always the first of the building blocks in a Conditional Assumption and there can be only one IF block per condition.

The IF block is made of at least two clauses:

- The IF clause contains the first logic clause of the block. For a Conditional Assumption, there can be only one IF clause.
- The THEN clause is the last clause of the block. You would insert a THEN clause when you have completed the definition of the logic for the block and are ready to assign a Transfer Pricing or Prepayment Method to those records that satisfy the logic.

In addition to the IF and THEN clauses, there are two other clauses you can optionally use to add to the complexity of the IF Block: the AND clause and the OR clause. They are always placed between the IF and the THEN clauses and there is no restriction on the order or amount that you can use.

The ELSE IF Block

The ELSE IF block serves as an alternative to any of the previous logic in a Conditional Assumption. It is used to add new blocks of logic within a Conditional Assumption, thus allowing for definition of more complex logic. There is no limit to the number of ELSE IF blocks you could add to a Conditional Assumption. Besides, their use is optional as the system does not require an ELSE IF block to build a valid Conditional Assumption.

The structure of the ELSE IF block is very similar to the IF block. The first clause indicates the starting point of the logic of the block; the IF and the ELSE IF have the same purpose, except that you can have only one IF clause in an entire Conditional Assumption but unlimited number of ELSE IF clauses. The last clause of the ELSE IF block must also be the THEN clause; it associates the logic of the particular block to a

Transfer Pricing or Prepayment Method. Like the IF block, you can have as many AND/OR clauses between the ELSE IF clause and the THEN clause.

The ELSE Block

The ELSE block is the last component of a Conditional Assumption. This block has only one clause. It is used to assign a Transfer Pricing or Prepayment Method to a record of the product-currency combination that does not satisfy the logic defined in all of the previous IF and ELSE IF blocks. The ELSE block ensures that all the records within the product-currency combination defined by you are transfer priced.

Like the IF block there can only be one ELSE block in a Conditional Assumption.

This table illustrates a conditional assumption structure comprising the IF, ELSE IF, and the ELSE blocks.

Conditional Assumption Structure: A Sample

Logic Block	Clause	Account Table Field	Operator	Value	Transfer Pricing Method
IF	IF	Field	Operator	Value	Transfer Pricing or Prepayment Method assigned to any instrument that meets logic of the IF Block.
	AND	Field	Operator	Value	
	AND	Field	Operator	Value	
	OR	Field	Operator	Value	
	THEN				
ELSE IF	ELSE IF	Field	Operator	Value	
	AND	Field	Operator	Value	

Logic Block	Clause	Account Table Field	Operator	Value	Transfer Pricing Method
	THEN				Transfer Pricing or Prepayment Method assigned to any instrument that meets logic of this ELSE IF Block.
ELSE IF	ELSE IF	Field	Operator	Value	
	AND	Field	Operator	Value	
	AND	Field	Operator	Value	
	OR	Field	Operator	Value	
	THEN				Transfer Pricing or Prepayment Method assigned to any instrument that meets logic of this ELSE IF Block.
ELSE	ELSE				Transfer Pricing or Prepayment Method assigned to any instrument that does not meet the logic of any of the previous blocks.

Related Topics

Setting Transfer Pricing Rules, page 2-3

Defining Transfer Pricing Methodologies, page 12-3

Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21

Defining Transfer Pricing Methodologies Using Node Level Assumptions, page 2-19

Setting Prepayment Rules

One of the major business risks faced by financial institutions engaged in the business of lending is the prepayment risk. Prepayment risk is the possibility that borrowers might choose to repay part or all of their loan obligations before the scheduled due dates. Prepayments can be made by either accelerating principal payments or refinancing.

Prepayments cause the actual cash flows from a loan to a financial institution to be different from the cash flow schedule drawn at the time of loan origination. This difference between the actual and expected cash flows undermines the accuracy of transfer prices generated using cash flow based transfer pricing methods.

Consequently, a financial institution needs to predict the prepayment behavior of instruments so that the associated prepayment risk is taken in to account while generating transfer rates. Oracle Transfer Pricing allows you to do this through the Prepayment Rule.

A Prepayment Rule contains methodologies to model the prepayment behavior of various amortizing instruments and quantify the associated prepayment risk. See: Prepayment Methodologies and Rules, page 2-27.

You need to first create rules and then versions. Prepayment methodologies are associated with the product-currency combinations within the versions of the Prepayment rule. See: Prepayment Rules, page 13-1.

Oracle Transfer Pricing allows you to make use of the node level and conditional assumption while defining prepayment methodologies for your products. See: Associating Node Level and Conditional Assumptions with Prepayment Rules, page 2-35.

Important: Prepayment assumptions are used in combination with only the three cash flow based transfer pricing methods: Cash Flow Weighted Average, Cash Flow Duration, and Cash Flow Zero Discount.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Defining Prepayment Methodologies, page 13-3

Prepayment Methodologies and Rules

You can use any of the following three methods in a Prepayment rule to model the prepayment behavior of instruments:

- Constant Prepayment method, page 2-28
- Prepayment Table method, page 2-28
- Arctangent method, page 2-31

Related Topics

Defining Prepayment Methodologies, page 13-3

Setting Prepayment Rules, page 2-27

Constant Prepayment Method

The Constant Prepayment method calculates the prepayment amount as a flat percentage of the current balance.

You can create your own origination date ranges and assign a particular prepayment rate to all the instruments with origination dates within a particular origination date range.

Note: All prepayment rates should be input as annual amounts.

Related Topics

Prepayment Methodologies and Rules, page 2-27

Defining Prepayment Methodologies, page 13-3

Defining the Constant Prepayment Method, page 13-6

Prepayment Table Method

The Prepayment Table method allows you to define more complex prepayment assumptions compared to the other prepayment methods. Under this method, prepayment assumptions are assigned using a Prepayment table.

You can build a Prepayment table using a combination of up to three prepayment drivers and define prepayment rates for various values of these drivers. Each driver maps to an attribute of the underlying transaction (age/term or rate) so that the cash flow engine can apply a different prepayment rate based on the specific characteristics of the record.

Note: All prepayment rates should be input as annual amounts.

Prepayment Table Structure

A typical Prepayment table structure includes the following:

- **Prepayment Drivers:** You can build a Prepayment table using one to three prepayment drivers. A driver influences the prepayment behavior of an instrument and is either an instrument characteristic or a measure of interest rates.
- **The Prepayment Driver Nodes:** You can specify one or more node values for each of the prepayment drivers that you select.
- **Interpolation or Range method:** Interpolation or Range methods are used to calculate prepayment rates for the prepayment driver values that do not fall on the defined prepayment driver nodes.

Types of Prepayment Drivers

The prepayment drivers are designed to allow the calculation of prepayment rates at run time depending on the specific characteristics of the instruments for which cash flows are being generated. Although nine prepayment drivers are available, a particular prepayment table can contain only up to three prepayment drivers.

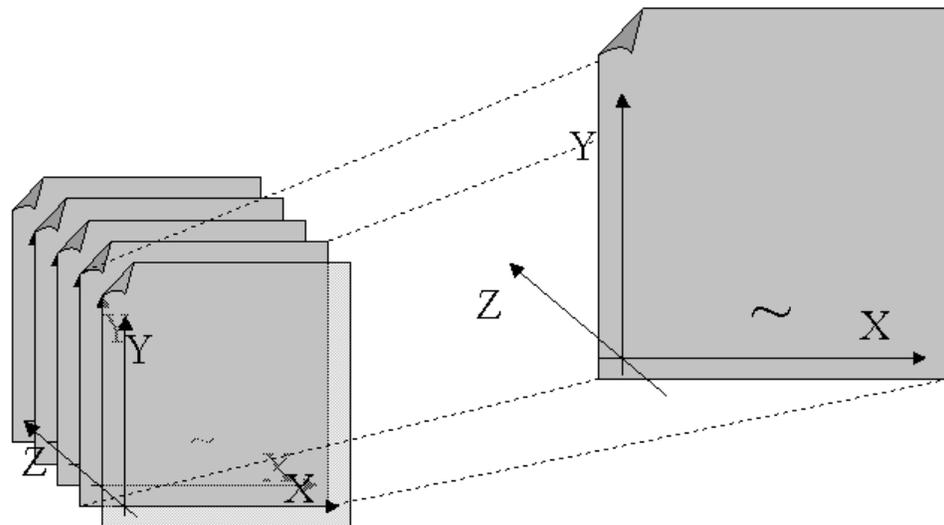
The prepayment drivers can be divided into the following two categories:

- **Age/Term Drivers:** The Age/Term drivers define term and repricing parameters in a Prepayment table. All such prepayment drivers are input in units of months. These Drivers include:
 - **Original Term:** You can vary your prepayment assumptions based on the contractual term of the instrument. For example, you could model faster prepayment speeds for longer term loans, such as a 10-year loan, than for short term loans, such as a 5-year loan. You would then select the Original Term prepayment driver and specify two node values: 60 months and 120 months.
 - **Repricing Frequency:** You can vary your prepayment assumptions based on the repricing nature of the instrument being analyzed. Again, you could specify different prepayment speeds for different repricing frequencies and the system would decide which one to apply at run time on a record by record basis.
 - **Remaining Term:** You can specify prepayment speeds based on the remaining term to maturity. For example, loans with few months to go until maturity tend to experience faster prepayments than loans with longer remaining terms.
 - **Expired Term:** This is similar to the previous driver but instead of looking at the term to maturity, you base your assumptions on the elapsed time. Prepayments show some aging effect such as the loans originated recently experiencing more prepayments than older ones.
 - **Term to Repricing:** You can also define prepayment speeds based on the number of months until the next repricing of the instrument.
- **Interest Rate Drivers:** The Interest Rate drivers allow the forecasted interest rates to

drive prepayment behavior to establish the rate-sensitive prepayment runoff. Interest Rate Drivers include:

- **Coupon Rate:** You can base your prepayment assumptions on the current gross rate on the instrument.
- **Market Rate:** This driver allows you to specify prepayment speeds based on the market rate prevalent at the time the cash flows occur. This way, you can incorporate your future expectations on the levels of interest rates in the prepayment rate estimation. For example, you can increase prepayment speeds during periods of decreasing rates and decrease prepayments when the rates go up.
- **Rate Difference:** You can base your prepayments on the spread between the current gross rate and the market rate.
- **Rate Ratio:** You can also base your prepayments on the ratio of current gross rate to market rate.

The following diagram illustrates a three-driver prepayment table:



The ~ signifies a point on the X-Y-Z plane. In this example it is on the second node of the Z-plane. The Z-plane behaves like layers.

Oracle Transfer Pricing allows you to build prepayment tables using the Prepayment Table rule. The Prepayment Table rule can then be referenced by a Prepayment Rule. See: Prepayment Table Rules, page 14-1.

Related Topics

Prepayment Methodologies and Rules, page 2-27

Defining Prepayment Methodologies, page 13-3

Arctangent Calculation Method

The Arctangent Calculation method uses the Arctangent mathematical function to describe the relationship between prepayment rates and spreads (coupon rate less market rate).

Note: All prepayment rates should be input as annual amounts.

User defined coefficients adjust this function to generate differently shaped curves. Specifically:

$$CPR_t = k_1 - (k_2 * ATAN(k_3 * (-C_t/M_t + k_4)))$$

where CPR_t = annual prepayment rate in period t

C_t = coupon in period t

M_t = market rate in period t

$k_1 - k_4$ = user defined coefficients

A graphical example of the Arctangent prepayment function is shown below, using the following coefficients:

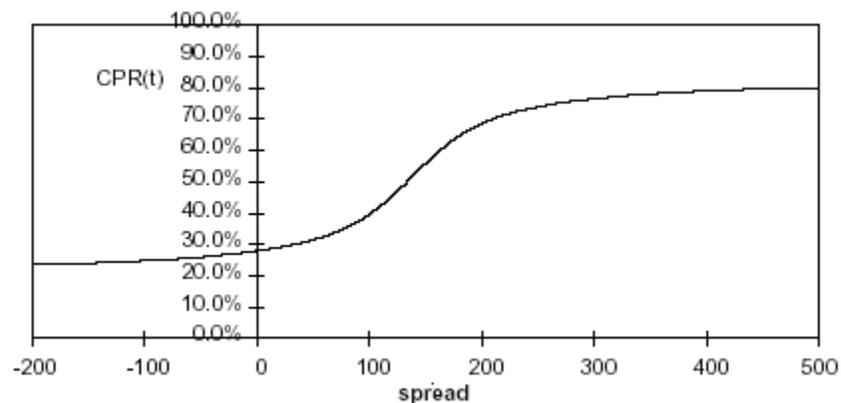
$$k_1 = 0.3$$

$$k_2 = 0.2$$

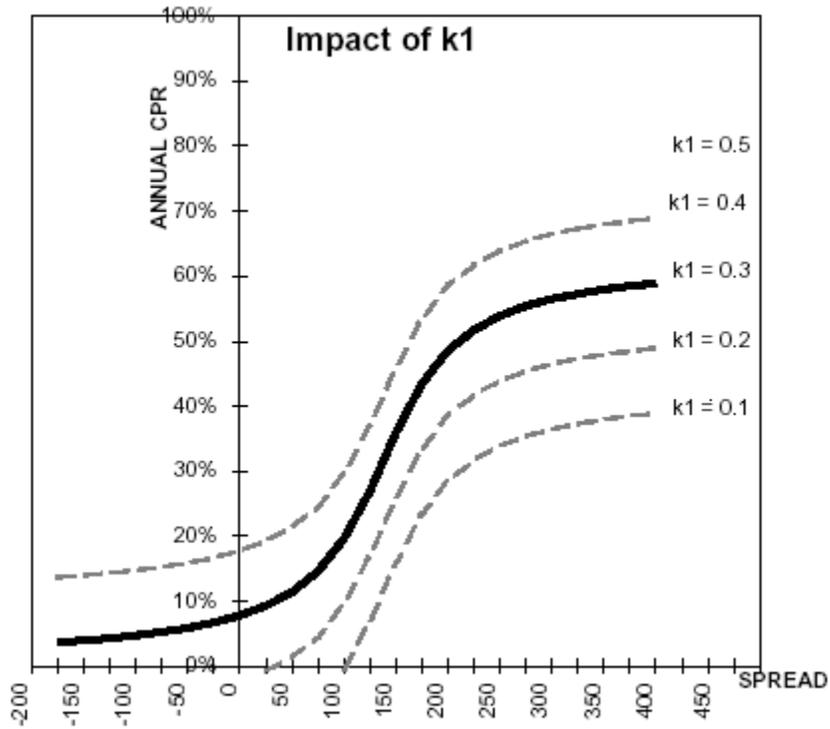
$$k_3 = 10.0$$

$$k_4 = 1.2$$

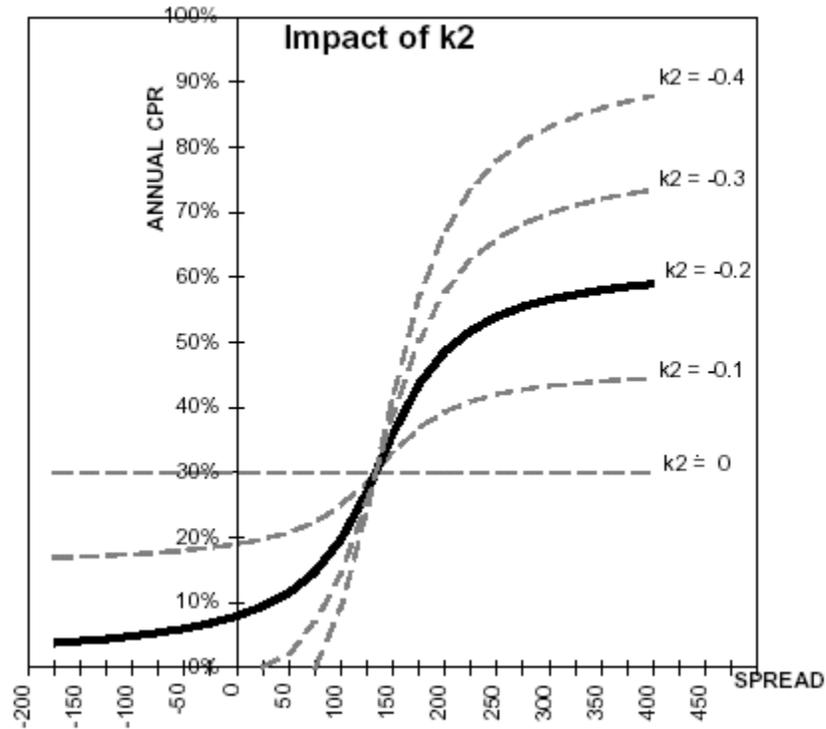
Each coefficient affects the prepayment curve in a different manner.



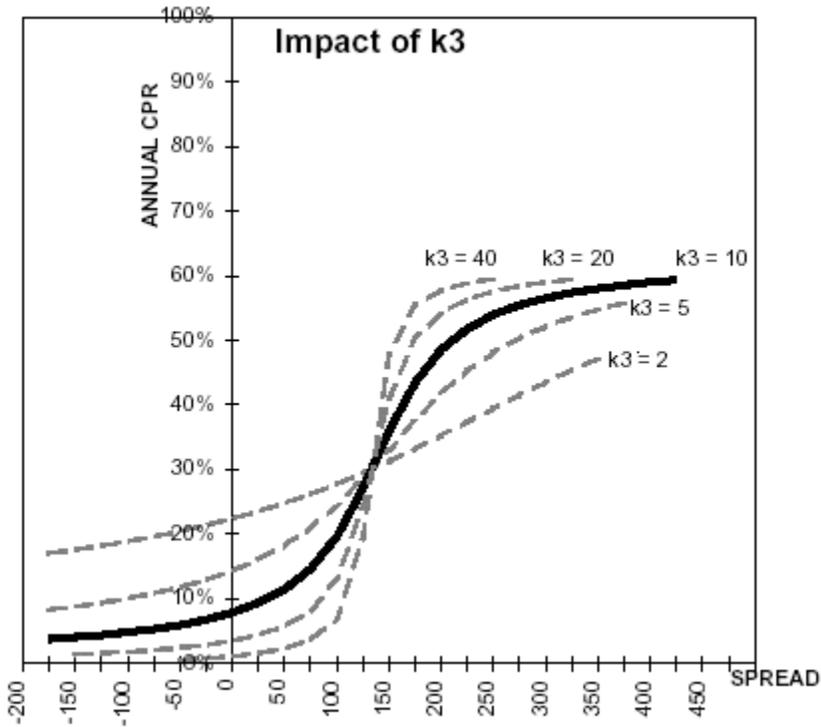
The following diagram shows the impact of K1 on the prepayment curve. K1 defines the midpoint of the prepayment curve, affecting the absolute level of prepayments. Adjusting the value creates a parallel shift of the curve up or down.



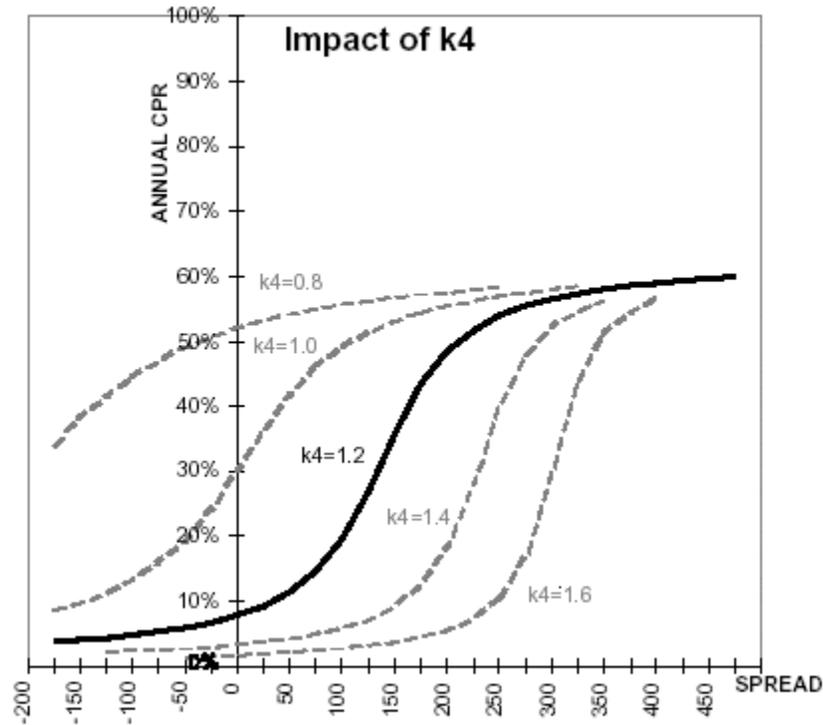
The following diagram shows the impact of K2 on the prepayment curve. K2 impacts the slope of the curve, defining the change in prepayments given a change in market rates. A larger value implies greater overall customer reaction to changes in market rates.



The following diagram shows the impact of K3 on the prepayment curve. K3 impacts the amount of torque in the prepayment curve. A larger K3 increases the amount of acceleration, implying that customers react more sharply when spreads reach the hurdle rate.



The following diagram shows the impact of K4 on the prepayment curve. K4 defines the hurdle spread: the spread at which prepayments start to accelerate. When the spread ratio = k4, prepayments = k1.



Related Topics

Prepayment Methodologies and Rules, page 2-27

Defining Prepayment Methodologies, page 13-3

Defining the Arctangent Method, page 13-9

Associating Node Level and Conditional Assumptions with Prepayment Rules

You can define prepayment methodologies at any level of the product hierarchy. Children of a hierarchical node automatically inherit the assumptions defined at the parent level. Methodologies directly defined for child nodes take precedence over those defined at the parent level. See: Defining Transfer Pricing Methodologies Using Node Level Assumptions, page 2-19.

You can also use the IF-THEN-ELSE logic of Conditional Assumptions to define prepayment methodologies based on particular characteristics of financial instruments. See: Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21.

Related Topics

Setting Prepayment Rules, page 2-27

Setting and Executing the Transfer Pricing Process Rule

Setting and executing the Transfer Pricing Process rule is the one of the mandatory steps in the Oracle Transfer Pricing process. The Transfer Pricing Process rule allows you to:

- Submit transfer pricing and prepayment assumptions, respectively contained in the associated Transfer Pricing and Prepayment rules, to the Transfer Pricing Cash Flow engine for processing.
- Determine the data that you want to process in a particular run.
- Define parameters used in transfer rate and option cost processing, migration of transfer pricing results to the Management Ledger table, propagation, and auditing.
- Choose the calculation mode for generating transfer pricing results, such as Remaining Term or Standard.
- Select which calculations, such as, Transfer Rates or Option Costs, or both, should be performed.

Setting the Transfer Pricing rule is a two-step process. You need to first create a rule and then a version. The Transfer Pricing Process rule has a separate Run page. Once a Transfer Pricing Process rule has been created, a user may select Run to execute it. The Run page allows you to supply all the run time parameters to get the required results. See: Transfer Pricing Process Rules, page 16-1.

When a Transfer Pricing Process rule is executed, detail account records are processed and individual records are updated with the results of the transfer pricing process. These results are based on the process options selected.

The following table displays the Account table fields that may be updated as a result of transfer pricing processing when you select Account tables as the data source.

Account Table Fields Updated by Transfer Pricing Processing

Calculation Type	Calculation Mode	Account Table Field
Transfer Rates	Standard	Transfer_Rate and Matched_Spread
Transfer Rates	Remaining Term	Tran_Rate_Rem_Term
Option Costs	Standard	Historic_Static_Spread and Historic_OAS

Calculation Type	Calculation Mode	Account Table Field
Option Costs	Remaining Term	Cur_Static_Spread and Cur_OAS

Additionally, you may choose the Management Ledger table as the data source for transfer pricing certain products. If the Management Ledger table data source is selected, the following rows are created for each product:

- Financial Element 170, Average Transfer Rate
- Financial Element 450, TP Charge/Credit

Important: Financial Element 140, Average Balance must exist in the Management Ledger table in order to successfully transfer price ledger balances.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Setting Transfer Pricing Rules, page 2-3

Transfer Pricing Rules, page 12-1

Setting Prepayment Rules, page 2-27

Prepayment Rules, page 13-1

Transfer Pricing Process Rule and Option Costs, page 2-37

Transfer Pricing Option Cost, *Oracle Financial Services Reference Guide*

Transfer Pricing Process Rule and Rate Index Rules, page 2-38

Rate Index Rules, page 10-1

Transfer Pricing Process Rule and Propagation Patterns, page 2-39

Propagation Patterns, page 15-1

Transfer Pricing Process Rule and Audit Options, page 2-39

Transfer Pricing Process Rule and Option Cost Parameters

In addition to transfer rates, the Transfer Pricing Process rule allows you to calculate the cost of options that are associated with the instruments. If you want to calculate option costs, you need to define the parameters used in options costing on the Transfer Pricing Process Version definition page. See: Transfer Pricing Process Rules, page 16-1.

The purpose of option cost calculations is to quantify the cost of optionality, in terms of a spread over the transfer rate, for a single instrument. The cash flows of an instrument with an optionality feature change under different interest rate environments and thus should be priced accordingly.

For example, many mortgages may be prepaid by the borrower at any time without penalty. In effect, the lender has granted the borrower an option to buy back the mortgage on par, even if interest rates have fallen in value. Thus, this option has a cost to the lender and should be priced accordingly.

In another case, an adjustable rate loan may be issued with rate caps (floors) which limit its maximum (minimum) periodic cash flows. These caps and floors constitute options.

Such flexibility given to the borrower raises the bank's cost of funding the loan and will affect the underlying profit. The calculated cost of these options may be used in conjunction with the transfer rate to analyze profitability.

Oracle Transfer Pricing uses the Monte Carlo technique to calculate the option cost. Oracle Transfer Pricing calculates and outputs two spreads and the option cost is calculated indirectly as a difference between these two spreads. These two spreads are:

- Static spread
- Option-adjusted spread (OAS)

The option cost is derived as follows:

- Option cost = static spread - OAS

The static spread is equal to margin and the OAS to the risk-adjusted margin of an instrument. Therefore, the option cost quantifies the loss or gain due to risk. See: *Transfer Pricing Option Cost, Oracle Financial Services Reference Guide*.

Related Topics

Setting and Executing the Transfer Pricing Process Rule, page 2-36

Transfer Pricing Process Rule and Rate Index Rules

The Rate Index rule is one of the parameters that you need to define on the Transfer Pricing Process Version definition page to calculate option costs. See: *Transfer Pricing Process Rules, page 16-1*.

The purpose of the Rate Index Rule is to establish a relationship between a risk-free Interest Rate Code (IRCs) and other interest rate codes or Indexes. The Rate Index rule allows you to select the valuation curve that the system uses during stochastic processing. The Rate Index rule provides full support for multi-currency processing by allowing you to select one valuation curve per currency supported in your system. See: *Rate Index Rules, page 10-1*.

During the stochastic processing, the system generates future interest rates for the

valuation curve you selected, which is then used to derive the future interest rates for any Index associated to that valuation curve based on the relationship you define. The rates thus forecasted for the IRCs or Indexes depend on the risk-free curve used for valuation of instruments associated with the derived IRCs or Indexes. As the risk-free rates change, the non risk-free interest rates change accordingly.

Related Topics

Setting and Executing the Transfer Pricing Process Rule, page 2-36

Transfer Pricing Process Rule and Propagation Pattern

Transfer Pricing theory suggests that a Fixed Transfer Rate should apply to an instrument record throughout its entire life (for Fixed Rate Instruments) or Repricing Term (for Adjustable Rate Instruments).

Propagation Patterns allow you to move forward (propagate) the Transfer Rate and Matched Spread on any applicable instrument record from a prior period of history. Propagation methodologies are system specific and can be used across process rules. See: Propagation Patterns, page 15-1.

The Transfer Pricing Process Rule allows you to propagate Transfer Rates as well as Options Costs. Depending upon your requirements, you can choose to propagate either the transfer rate or the option cost or both on the Transfer Pricing Process Run page. See: Transfer Pricing Process Rules, page 16-1.

The main goal of using propagation is to increase performance. Since Propagation uses a bulk processing approach, it provides a significant performance improvement over processing instruments with a row-by-row approach. Although precise performance numbers may vary depending on the hardware and database configuration, processing a set of instrument records using propagation is significantly faster than doing it on the same set of records on a row-by-row basis.

Related Topics

Setting and Executing the Transfer Pricing Process Rule, page 2-36

Transfer Pricing Process Rule and Audit Options

The Transfer Pricing Process rules provides you with the three audit options: Detailed Cash Flow, Forward Rates, and 1 Month Rates. While Detailed Cash Flow audit option is applicable to both transfer rate and option cost processing, the Forward Rates and 1 Month Rates audit options are applicable only to option cost processing.

By selecting the Detail Cash Flow option in the Transfer Pricing Process Rule Run flow, you can audit daily cash flow results generated by the Oracle Transfer Pricing application. Selecting this option writes out all cash flow and repricing events that occur for processed records. The number of records written is determined by the environment on which the process is running. If you are running under multiprocessing, you may

get fewer records, for example,
OFSA_PROCESS_ID_STEP_RUN_OPT.NUM_OF_PROCESSES > 1.

The relevant financial elements for each instrument record and the cash flow results are stored in the FTP_PROCESS_CASH_FLOWS table. See: FTP_PROCESS_CASH_FLOWS, page 2-55.

After processing cash flows from Transfer Pricing, do the following to view the audit results:

- Determine the value of Object ID. See: Executing a Transfer Pricing Process Rule, page 16-9.
- View data by:
 - Creating a Condition associated with the Object ID Number. See: About Conditions, *Oracle Enterprise Performance Foundation User's Guide*.
 - Querying the FTP_PROCESS_CASH_FLOWS table with a Data Inspector rule to view the data. See: About the Data Inspector and Data Inspector Rules, *Oracle Enterprise Performance Foundation User's Guide*.

Note: Oracle Transfer Pricing has a seeded Oracle Discoverer workbook to help you query the cash flow audit results.

Related Topics

Setting and Executing the Transfer Pricing Process Rule, page 2-36

Data Inspector Results

The results of running the Data Inspector appear in a spreadsheet.

Financial Elements

The FINANCIAL_ELEMENT_ID column lists the financial elements written for each payment and repricing event processed by the cash flow engine. An initial set of data is also written, recording the balance and rate as of the last payment date.

The following table describes the financial elements that can be present in a base set of financial elements written during a cash flow audit process:

Financial Elements Written During Audit

Financial Element	Description
Initial Event	

Financial Element	Description
100	Ending par balance. Final balance on payment date, after the payment has occurred.
430	Interest cash flow.
210	Total principal runoff, including scheduled payments, prepayments, and balloon payments.
60	Beginning par balance. Starting balance and payment date, prior to payment.
120	Runoff Net Rate. Rate at the time of payment, weighted by ending balance. To view actual rate, divide financial element 120 by financial element 100.
490	(Stochastic Processes Only) Discount factor used during Monte Carlo process to determine present value of cash flow on the payment date.
Initial Event	
250	Par balance at time of repricing.
280	Before Reprice Net Rate. Rate prior to repricing, weighted by reprice balance. To determine true rate, divide this financial element value by financial element 250.
290	After Reprice Net Rate. Newly assigned net rate after repricing occurs, weighted by reprice balance. To determine true rate, divide this financial element value by financial element 250.
Initial Event	
100	Initial par balance at start of processing.
120	Initial net rate at start of processing.

In addition to these financial elements, other data may be output depending on the type of processing and the optional financial elements selected.

Cash Flow Codes

The Cash Flow Code column lists a code for each row that describes the event modeled by the cash flow engine.

The following tables describes the different cash flow codes:

Description of Cash Flow Codes

Cash Flow Code	Description
1	Initial recording of balances and rates.
2	Payment event only.
20	Reprice event only (not during tease period).
8	Reprice during tease period.
22	Reprice and payment event together (not during tease period).
10	Reprice and payment during tease period.

Data Verification

You can copy the results from the Process Cash Flows table and paste them into a spreadsheet to facilitate analysis against validated data. If the cash flows do not behave as expected, examine instrument table data or your assumptions. See: Cash Flow Calculations, *Oracle Financial Services Reference Guide*.

Transfer Pricing Process Rule and Calculation Modes

You can choose to transfer price your product portfolio either in the Standard or in Remaining Term calculation mode.

The Standard calculation mode allows you to calculate transfer rates for instrument records based on the Origination date or Last Repricing Date of the instruments. It can also be used to calculate Option Costs based on the Origination Date.

The Remaining Term calculation mode allows you to calculate transfer rates and option costs for instrument records based on the remaining term of the instrument from the As of Date of the data, rather than the Origination Date or Last Repricing Date of the

instruments.

The Remaining Term calculation mode treats your portfolio as if you acquired it on the As of Date of your data and thereby allows you to measure current rate risk spread. Once you know the current rate risk spread, you can segregate your total rate risk spread into that accruing from taking current rate risk and that accruing from taking embedded rate risk:

$$\text{Embedded Rate Risk Spread} = \text{Total Rate Risk Spread} - \text{Current Rate Risk Spread}$$

It is important to segregate total rate risk into embedded and current rate risks for the following reasons:

- The current rate risk can be actively managed through an effective Asset/Liability Management process.
- Embedded rate risk is a result of rate bets taken in the past. However, it is important to measure and monitor this risk. When you are aware of your embedded rate risk you will neither be lulled into a false sense of security or take drastic actions in response to profit or losses caused by the embedded rate risk.

See: Evaluating Interest Rate Risk, *Oracle Financial Services Reference Guide*.

Transfer Pricing Process Rule and Migration Options

The purpose of the Ledger Migration process is to generate dollar credits or charges for funds provided or used for a combination of dimensions. The information necessary to generate these credits or charges (through transfer rates and option cost processing) originates from the Customer Account Tables and the results are inserted into the Management Ledger table, and are available for use in calculation of profitability and risk measures.

Note: The Management Ledger table is also known as the FEM_BALANCES and the Customer Account tables are also known as Instrument tables.

Oracle Transfer Pricing provides great flexibility in the ledger migration process and the generation of corresponding charges, credits, and option costs. Users can specify ledger migration for a combination of an extended list of dimensions, including up to 10 user-defined dimensions. This feature enables the system with profitability reporting capabilities across organizational, product, channel, geography, and user defined dimensions.

In addition, Oracle Transfer Pricing provides multi-currency support that allows you to generate charges or credits for funds based on entered and functional currency. See: The Ledger Migration Process, *Oracle Financial Services Reference Guide*.

You can choose to migrate either the transfer rate or the option costs, or both, on the Transfer Pricing Process rule run page. See: Transfer Pricing Process Rules, page 16-1.

Related Topics

Setting and Executing the Transfer Pricing Process Rule, page 2-36

Defining Payment Patterns

A prerequisite for transfer pricing your product portfolio is capturing instrument behavior, such as payment and repricing patterns. This is because transfer rate and option cost processing require cash flow generation, which is possible only if you have accurately captured the payment and repricing profiles of the products in your portfolio.

The payment and repricing patterns for most instruments can be accommodated in the Account tables. However, certain instruments may have payment and repricing patterns that are too complex to be accommodated in the standard fields of Account tables. Oracle Transfer Pricing allows you to define custom payments and repricing patterns for such instruments. See: User Defined Payment Patterns, page 7-1 and User Defined Repricing Patterns, page 8-1.

In a user defined payment pattern, you can assign a unique amortization code to a set of payment events, which may include some of the following customized features:

- Changes in payment frequency
- Seasonal payment dates
- Nonstandard or variable payment amounts

Once you create a payment pattern, you can use it by entering the payment pattern code as the amortization type code for the instrument.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Defining Repricing Patterns, page 2-48

Payment Pattern Structure

Oracle Transfer Pricing allows you to build three types of payment patterns:

- Absolute, page 2-47
- Relative, page 2-48
- Split, page 2-48

These payment patterns differ in terms of how they address payment schedules, which determine whether the payment events constituting the pattern are determined by

calendar dates or periods. Absolute patterns are defined with sets of payment characteristics scheduled on specific calendar dates. Relative patterns are defined with sets of payment characteristics scheduled for certain periods of time.

You can also define a payment pattern with both absolute and relative payment events. This type of pattern is called a split pattern.

In addition, for each payment pattern, you need to specify a payment type, either conventional, level principal, or non-amortizing. Your choice of the pattern type and the payment types will determine the fields that are used for calculation.

Note: Oracle Transfer Pricing's Payment Pattern interface supports simultaneous multiple-user access.

Related Topics

Defining Payment Patterns, page 2-44

User Defined Payment Patterns, page 7-1

Payment Events

You must define one or more payment events to complete a payment pattern. A payment event is a set of payment characteristics, which define the time line and amount of a specific payment in the payment pattern.

Though the characteristics of the payment phase change based on whether you are defining an absolute, relative, or split pattern, there are two characteristics that are required for all patterns:

- Payment method
- Value

Payment Method

The payment methods determine the payment amount for the payment event. There are six different methods.

The following table describes the different payment methods.

Payment Methods

Method	Description
% of Original Balance	This method calculates the payment as a percentage of the original balance; the percentage being defined by the input percent. This method is useful for apportioning the starting balance on a level principal instrument over several payments. This method is only available for payment patterns defined with a level principal payment type.
% of Current Balance	This method calculates the payment as a percentage of the current balance prior to payment; the percentage being defined by the input percent. This method is only available for payment patterns defined with a level principal payment type.
% of Original Payment	This method calculates the payment as a percentage of the original payment column from the detail instrument data. This percentage is defined by the input percent.
% of Current Payment	This method calculates the payment as a percentage of the previous payment; the percentage being defined by the input percent. This payment is calculated on the payment date based on the characteristics of the instrument at the time of the payment, including the current rate, current balance, and current payment frequency.
Absolute Payment	This is an input payment amount. This amount represents both principal and interest for a conventional payment type, and represents only principal for a level principal payment type. For both types of patterns, absolute value payment amounts are entered as gross of participations.
Interest Only	This is an input payment amount. An interest-only payment is calculated during processing as balance times rate times accrual factor.

Value

The value reflects the percentage or payment amount based on the method chosen for the payment event. Value is disabled for phases using the Interest Only payment method.

Payment amounts for conventional pattern phases must reflect both principal and interest payments. Payment amounts for level principal pattern phases only reflect the principal portion of the payment. For level principal pattern phases, the total cash flow on a payment date is the principal amount stored as the payment plus the calculated interest.

Note: The payment method and value columns are not displayed for payment patterns defined with a non-amortizing payment type. All payments are assumed to be interest only for this type of payment pattern.

Related Topics

Defining Payment Patterns, page 2-44

User Defined Payment Patterns, page 7-1

Absolute Payment Patterns

Absolute payment patterns are commonly used for instruments that are on a seasonal schedule, such as agricultural or construction loans that require special payment handling based on months or seasons.

Take the example of a loan that follows a seasonal payment pattern, in which the payment patterns for January, February and March are scheduled for interest-only payments. As revenues for the customer increase, the payment amount also increases. Therefore, the payments for April and May are 80% of the original payment, and June through September is 100% of the original payment. The payment decreases as the production season slows. The payment for October is decreased to 80% of the original payment, and the payments for November and December are decreased again to 50% of the original payment.

See: Defining Absolute Payment Patterns, page 7-4.

Note: You can define absolute payment patterns only up to a year. This is because all entries are automatically ordered by date order and are scheduled in a single year rotation.

Related Topics

Defining Payment Patterns, page 2-44

User Defined Payment Patterns, page 7-1

Relative Payment Patterns

Relative payment patterns are commonly used for modeling instruments with irregular payment frequencies or for instruments where the payment type changes over time.

Take the case of a four-year loan for example. The payment for the first 12 months could only be interest. The first 35 payments are scheduled for 50% of the currently scheduled payment, and the last payment is a balloon payment for the balance of the loan.

See: Defining Relative Payment Patterns, page 7-7.

Related Topics

Defining Payment Patterns, page 2-44

User Defined Payment Patterns, page 7-1

Split Payment Patterns

A split pattern contains multiple sets of payment patterns under a single amortization code. You use a split pattern for financial instruments that make principal payments along two concurrent amortization schedules. Each separate amortization schedule is termed a time line and assigned a percentage of the balance. A Split Pattern can constitute both absolute and/or relative payment patterns within itself. See: Defining a Split Payment Pattern, page 7-10.

Related Topics

Defining Payment Patterns, page 2-44

User Defined Payment Patterns, page 7-1

Defining Repricing Patterns

User defined repricing patterns provide a mechanism to capture the repricing structure of instruments whose rates change according to complex schedules which can not be captured in the standard fields of account tables. See: User Defined Repricing Patterns, page 8-1 and User Defined Payment Patterns, page 7-1.

The user defined repricing pattern allows you to define multiple changes to various elements affecting repricing including:

- Rates
- Margins
- Frequency

A repricing pattern has two major components:

- User Defined Repricing Pattern, page 2-49
- User Defined Repricing Event, page 2-49

Note: Oracle Transfer Pricing Repricing Pattern interface supports simultaneous multiple-user access.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Defining Payment Patterns, page 2-44

User Defined Repricing Pattern

The user defined repricing pattern provides you with the ability to define a series of repricing patterns and events that describe the interest rate adjustment characteristics over the life of a cash flow instrument. One repricing pattern can be assigned to many cash flow instruments.

There are two types of repricing patterns that you can define:

- Absolute Repricing Pattern, page 2-51
- Relative Repricing Pattern, page 2-52

See: Creating a Repricing Pattern, page 8-3.

Related Topics

Defining Repricing Patterns, page 2-48

User Defined Repricing Patterns, page 8-1

User Defined Repricing Event

The events of a repricing pattern define changes to the interest rates of an instrument during its life. Every pattern begins with an initial event, which describes the behavior for the initial period.

Note: This initial event is required for the setup of all repricing patterns but is not used in Oracle Transfer Pricing. This feature is used only by Risk Manager, another Oracle Financial Services application, when assigning a rate at origination of new business and transaction strategy records.

The second event describes the change in behavior after the initial period is over. A third event describes the next change in behavior and so on. In relative repricing patterns, you can also define the number of times an event will be repeated before the next event is triggered.

At least one event must be defined for a repricing pattern. All events are listed in the Repricing Events table. The repricing pattern type, absolute or relative, determines the data required to be populated in the events table.

Caution: You have the option to change the repricing pattern type at any time during the create process. However, changing the repricing pattern type causes the system to automatically refresh the Repricing Events table, and the loss of all the data that you previously entered.

Event Detail

You define each event with a repricing type of either flat rate or indexed rate. The repricing types determine the event detail characteristic that are available.

Flat Rate

Selecting the flat rate repricing type allows you to set the rate of the instrument to a fixed value. For example, 6%.

The following table describes the event detail characteristics that are available when the flat rate repricing type is selected.

Event Detail Characteristics: Flat Rate

Characteristic	Description
Net Rate	The new net rate value
Gross Rate	The new gross rate value
Transfer Rate	The new transfer rate

Flat rate always overrides the caps and floors defined on the instrument record.

Indexed Rate

Selecting the indexed rate repricing type allows you to set the rate of the instrument to an adjustable value, defined as the index rate plus a margin.

The following table describes the event detail characteristics that are available when the indexed rate repricing type is selected:

Event Detail Characteristics: Indexed Rate

Characteristic	Description
IRC (Interest Rate Code)	Reference interest rate used as the index rate to set gross and net rates. This list of values is pulled from the current Historical Rates database.
Transfer Rate IRC	Interest rate used to calculate transfer rate. The field is a list of value type.
Net Margin	Added to index rate to get net rate.
Gross Margin	Added to index rate to get gross rate.
Transfer Margin	Added to index rate to get transfer rate.
Rate Cap Life	The upper limit for gross rate set by a particular event.
Rate Floor Life	The lower limit for gross rate set by a particular event.
Rate Set Lag	Period by which the date of the interest rate used for calculation precedes the event date; set with a value and a multiplier.
Yield Curve Term	Term used in interest rate code lookups; if left blank, defaults to the term until the next repricing; set with a value and multiplier.

Related Topics

Defining Repricing Patterns, page 2-48

User Defined Repricing Patterns, page 8-1

Absolute Repricing Pattern

The absolute repricing pattern is used for instruments that are date dependent. Each specific date is a separate event.

You may have up to one year of defined events that repeat for the life of the instrument. For example, you could define one event for each day of the year; the maximum

number of events that you can define is 365. However, you can only define one event for any given date. See: Defining Absolute Repricing Pattern, page 8-4.

Related Topics

Defining Repricing Patterns, page 2-48

User Defined Repricing Patterns, page 8-1

Relative Repricing Pattern

The relative repricing pattern is a series of repricing events that are driven by user defined time lines. It is used for instruments where the repricing is determined by elapsed time since origination. You specify the duration of each repricing period (frequency) and the number of times that event should occur (repeat) before calculating the next event in the pattern.

For example, an event can be defined with a frequency of 1, a multiplier of Months, and a repeater of 3. This translates into an event that reprices every month for a duration of 3 consecutive months.

You may have a graduated rate mortgage that requires three rate changes over the life of the instrument. You will have three events following the initial event. If you wish the instrument to retain the behavior defined for the last event, the repeater should be set to 999. This prevents *wrapping*, or the repetition of all the defined events until the life of the instrument runs out. See: Defining Relative Repricing Pattern, page 8-7.

Related Topics

Defining Repricing Patterns, page 2-48

User Defined Repricing Patterns, page 8-1

Performing Cash Flow Edits

It is extremely important that the data in Account tables is clean, accurate, and complete before it is used to generate cash flows and for further processing. Oracle Transfer Pricing provides Cash Flow Edit rules to edit (clean and prepare) Account table data. You can create multiple Cash Flow Edit rules depending on the data to be cleansed. In addition, you can view actual results of Cash Flow Edits by accessing the result data written into the FTP_CF_CORRECTIONS table.

You can also select the preview option so that you can preview the changes that will be made to the Account table data as a result of cash flow edits before those changes are applied in the account tables.

It is highly recommended that you create and run Cash Flow Edit rules before processing data to generate any type of cash flow-related results. See: Cash Flow Edits Rules, page 6-1.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Creating Interest Rate Codes

Oracle Transfer Pricing uses historical interest rate information to transfer price your balance sheet. The final transfer rate assigned to the instruments in your account tables is based on the historical rates information stored in the system. Consequently, you must decide on the type and amount of historical rate information you require to satisfy your transfer pricing requirements at the outset of an Oracle Transfer Pricing implementation.

However, the quality and availability of interest rate information varies throughout the world. In many markets, gathering comprehensive rate information is a challenge because of insufficient security types, inconsistent quoting conventions, and lack of liquidity. This necessitates careful management of the interest rate data. In Oracle Transfer pricing, this is done using reference interest rates, called interest rate codes.

Creating interest rate codes is one of the mandatory steps in the Oracle Transfer Pricing process. Oracle Transfer pricing provides a separate rule, called Interest Rate Codes (IRC), to define and manage historical interest rates for transfer pricing purposes.

Oracle Transfer Pricing facilitates the process of inputting and viewing interest rates by giving you data storage capabilities appropriate to your market. This is possible as the application supports multiple rate formats and allows you to store the following rate attributes:

- Rate format (zero-coupon or yield-to-maturity)
- Accrual basis
- Compound basis

In addition to historical interest rate information, Oracle Transfer Pricing allows you to manage the term structure modeling parameters, such as volatility and mean reversion speed. See: Interest Rate Codes, page 9-1.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Interest Rate Codes and Rate Lookups

A rate lookup is performed to derive a transfer rate for the appropriate date/term combination.

Date Used: Oracle Transfer Pricing accesses the yield curve based on the appropriate

lookup date. If no match is found, it uses the first date before the date of your lookup.

Term Used: Oracle Transfer Pricing selects the term on the yield curve on an exact number of days basis, calculated by subtracting the cash flow date from the transfer pricing date, which may be the effective date, the last reprice date, or the origination date depending on the method and the instrument characteristics.

If the yield curve term is expressed in months or years, the term must be converted to a days basis, as follows:

If Multiplier = M (month), Term in Days = Term in Months * 30.42

If Multiplier = Y (year), Term in Days = Term in Years * 365

The rate is then derived from the yield curve by performing linear interpolation to the two points between which the lookup term falls.

Rate Lookup at Endpoints

If the term < shortest point on the yield curve, then the rate = the shortest point.

If the term > longest point on the yield curve, then the rate = the longest point.

If the date for the lookup > dates available then the lookup is on the last date for the yield curve.

If the date for the lookup < dates available then the lookup is on the first date for the yield curve.

Rate Lookup: An Example

The following table displays transfer rates for different date/term combinations:

Date	1 Day	1 Month	3 Months	1 Year
01/01/2004	2.00	3.00	4.00	5.00
01/15/2004	2.10	3.10	4.10	5.10
01/31/2004	2.20	3.20	4.20	5.20
02/15/2004	2.30	3.30	4.30	5.30

The following table displays Date/Term Combinations for Lookup:

Date	Lookup Term	Yield Curve Date Used	Term Before	Term After	Rate	Comments
01/07/2004	60 days	01/01/2004	1 Month	3 Months	3.50	Rate is approximately half way between 3 Months (91.26 Days) and 1 Month (30.42 Days).
11/30/2003	182 days	01/01/2004	3 Month	1 Year	4.33	3 Months Rate + (182 Days - 91.26 Days) * (1 Year Rate - 3 Months Rate) / (365 Days - 91.26 Days) (such as 1/3 of the Way between 3 Months and 1 Year).
03/15/2004	2 Year	02/15	1 Year	None	5.30	Uses last point on Yield Curve.

Accessing Transfer Pricing Detail Cash Flow Results for Audit Purposes

Detailed cash flow results for individual account records can be written to an audit table for validation purposes. If you select the Detailed Cash Flows audit option on the Transfer Pricing Process rule run page, the detailed cash flow results are written to the FTP_PROCESS_CASH_FLOWS table. The following table describes the columns that make up the FTP_PROCESS_CASH_FLOWS table:

Columns in FTP_PROCESS_CASH_FLOWS Table

Column Name	Description
OBJECT_ID	Key column that corresponds to the internal ID of the Transfer Pricing Process rule. All cash flow results for your processing run are stored with the same ID value.

Column Name	Description
RECORD_SEQUENCE	The processing order of the records.
CASH_FLOW_SEQUENCE	The order of the events, such as cash flows and repricing.
SCENARIO_NUM	The scenario number assigned in the forecast rates assumption (used in Risk Manager).
FINANCIAL_ELEM_ID	The numeric code describing a piece of financial information described in the row of results data
ID_NUMBER	The unique record identifier for the processed instrument records.
CASH_FLOW_DATE	The date of the event.
CASH_FLOW_CD	<p>The code identifying the event. The code can be any combination of the following base codes:</p> <ul style="list-style-type: none"> • 1=1st • 2=payment • 4=reprice • 8=in tease period • 16=not in tease period • 32=effective date. For example, 22 is the combination of 2,4, and 16.
FLOAT_VALUE	The value assigned to each financial element.

Column Name	Description
CALC_SOURCE_CODE	The type of cash flow processing: The code can be: <ul style="list-style-type: none"> • 0 - RM - Regular (scenario-based processing) • 1 - RM - Stochastic processing • 2 - TP - Regular (scenario-based processing) • 3 - TP - Forward Rates • 4 - TP - Stochastic processing
SOURCE_SYSTEM_CODE	Indicates the application from which the detail cash flow processing was initiated.
CURRENCY_CODE	The currency associated with each line item.
LINE_ITEM_ID	The Transfer Pricing Line Item dimension member. Represents the Product dimension.
COMPANY_COST_CENTER_ORG_ID	The organizational unit number.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Analyzing Results

You should always analyze results obtained from the Transfer Pricing engine. For example, you should review the historical rates information (Interest Rate Codes) to ensure that the new cost of funds reflect the current interest rate reality.

In addition, a detailed transfer rate/matched spread query should be generated at the product level to ensure that every account has been assigned a transfer rate and that the matched spread for each account is as expected.

The following table lists some steps to find out whether an account has not been transfer priced correctly:

Steps to Analyze Transfer Pricing Results

Query	Results
Stratification by transfer rate	Look for any transfer rate <= a selected value (such as 3.00) or >= another value (such as 12.00) look for any transfer rate <= a selected value (such as 3.00) or >= another value (such as 12.00)
Stratification by matched spread	Look for large (positive or negative) matched spreads. (for example, >= 4.00 or <= -2.00)
Stratification of fixed rate instruments by origination date and term with weighted average transfer rate and matched spreads as columns	Look for general pattern to reflect the Transfer Pricing Yield Curves for each last repricing date
Stratification of adjustable rate instruments by last repricing date and term with weighted average transfer rate and matched spreads as columns	Look for general pattern to reflect the Transfer Pricing Yield Curves for each last repricing date

In case a result (transfer rate) generated by the system is suspect, then you can view all of the cash flows for any specified instrument record, by selecting the Detailed Cash Flow option in the Transfer Pricing Process rule. This option should be selected together with a condition, which identifies the instruments to be included in the Audit process.

After ensuring that each account has been assigned an accurate transfer rate, you should review the funding center impact and compare it to the results from prior periods.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Reviewing Processing Errors

There is always the possibility that errors might occur during the execution of a Transfer Pricing Process rule. A log of such errors is generated during processing and can be accessed from within the concurrent manager. Within this log, the application identifies the specific transaction for which an error was generated and provides the internally generated identifier of the Transfer Pricing Process rule that generated it.

As part of the rectification process, it is advisable to determine what caused the error

and what should be done to correct it for the next run.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Reprocessing Erroneous Accounts

While reviewing your results, you might discover accounts with invalid results that need to be reprocessed. The Transfer Pricing Process rule allows you to rerun a subset of information.

If you need to reprocess a portion of your instrument data, make sure that you reprocess all the Line Item dimensions members for a rule. Otherwise, you may damage your overall results.

If any of the records being reprocessed are used as the basis for unpriced accounts, those unpriced accounts also should be reprocessed.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Reconciling the Data

Reconciliation is the process of comparing the information carried in the Account tables to the general ledger.

The goal of the Transfer Pricing Process rule is to transfer price your entire balance sheet, as represented on the general ledger. Many ledger accounts have corresponding data in the Account tables. In such instances, the balances from the instrument data must be compared with the corresponding ledger balances.

The reconciliation process involves defining a level at which some piece of information in the Account tables is to be compared to the General Ledger data carried in the Management Ledger (also known as Ledger Stat and FEM_BALANCES) table. That level can be one dimension (to reconcile for each general ledger account number, for example, Natural Account ID) or multiple dimensions (to reconcile for each general ledger account number within each business unit, for example, Natural Account and Company Cost Center Org ID).

The most common type of reconciliation is to compare the current balance of Account table data to the general ledger ending balance. The data carried in the database is a snapshot of the portfolio as of a given date. Consequently, comparing the current balances from the Account table to the general ledger ending balance measures the degree to which the extracted data is in balance with, or reconciles to, the general ledger.

Variations between the Account table and the Management Ledger table should be corrected. If the magnitude of the variations is high, plug entries should be created to force the reconciliation to zero.

Related Topics

Overview of the Oracle Transfer Pricing Process, page 2-1

Common Rule Management Tasks

This chapter focuses on the rule management tasks that are common across all rules in this application.

This chapter covers the following topics:

- Overview of Common Rule Management Tasks
- The Rule Home Page
- Searching for Rules
- Creating Rules
- Viewing and Updating Rules
- Duplicating Rules
- Deleting Rules
- Extracting and Loading Rules

Overview of Common Rule Management Tasks

The rule management tasks that are common to business rules in this and other applications are as follows.

- The Rule Home Page, page 3-2
- Searching for Rules, page 3-5
- Creating Rules, page 3-6
- Viewing and Updating Rules, page 3-7
- Duplicating Rules, page 3-7
- Deleting Rules, page 3-9

- Extracting and Loading Rules, page 3-9

Note: You can perform these tasks from the home page for the type of rule with which you are working. Depending on the rule type, some tasks might not be available.

The procedures for carrying out these tasks are the same for each rule type, except for rule-specific steps explicitly stated in the rule-specific documentation.

The Rule Home Page

The Rule home page is the gateway to all rules and related functionality of the application. From there, you can navigate to other related pages.

On the header of the Rule home page, you can perform simple queries on Folder, Rule Name and Effective Date.

The following table shows the page components.

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Folder	Drop Down	Set in Application Preferences	Required - for filtering the rules under the folder	No – Only able to select from presented list.	N/A
(Rule) Name	Text Box	None	Optional – for filtering the rules on Rule Name	Yes	You can specify all or part of a rule name For example, if you want to see only those Rules which start with 'A' – Enter A in the text field

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Effective Date	Date Field	None	Optional – for filtering the rules on Version Effective Dates	Yes	Returns those Rules with Version effective dates that correspond with the date you enter
Go	Button	N/A	N/A	No	Initiates rule search based on specified criteria.
Create Rule	Button	N/A	N/A	No	Initiate the Data or Ledger Loader rule creation process.
Expand All	Button	N/A	N/A	No	Displays rules with version information
Collapse All	Button	N/A	N/A	No	Displays only with the Rule names
Focus	Button	N/A	N/A	No	Displays the selected Rule's Version information will be shown
(Rule) Name	Link	N/A	N/A	No	Links to the Rule's basic details

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Start Date	Display Value	N/A	N/A	No	Version effective start date
End Date	Display Value	N/A	N/A	No	Version effective end date
Approval Status	Display Value	N/A	N/A	No	Displays whether this Rule version is approved, submitted, or new
Approved By	Display Value	N/A	N/A	No	Who approved the Rule version
Approved Date	Display Value	N/A	N/A	No	When was the rule version approved
Submit	Button	N/A	N/A	No	Click to have the rule version approved Approved rules can run against production or non-production datasets Unapproved rules can only run against non-production datasets.
Revert	Icon	N/A	N/A	No	

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Duplicate	Button	N/A	N/A	N/A	Initiates process for duplicating rules. Explained later in this document
Run	Button	N/A	N/A	N/A	Initiates process for running Rules Explained later in this document.

Searching for Rules

Search for a business rule to perform any of the following tasks:

- Update, duplicate, export, migrate, or delete existing rules or versions
- Create a new version
- Define methodologies for new products

Procedure:

1. Navigate to the rule home page, page 3-2 for the appropriate rule type.
2. Search for the rule, as follows:
 1. Select the folder in which the rule is stored.
 2. (Optional) Enter the name of the rule.
 3. (Optional) Select the effective date.
 4. Click Go.

Only rules that match the search criteria are displayed.

Please refer to Overview of Common Rule Management Tasks, page 3-1 for

more information.

Creating Rules

You create a rule to specify the way you want a particular task or business process to be carried out by the application. Creating a rule is a two-step process, in which you first specify the properties for the rule itself, and then specify the properties for the rule version.

Procedure to Create a Rule and Version:

1. Navigate to the home page, page 3-2 of the rule you want to create.
2. Click Create to display the rule definition page.
3. Select the folder in which you want to store the rule.
4. Enter a name for the rule.

Important: The name of a rule must be unique across all the rules and rule types in the entire database, not just at the folder level.

5. (Optional) Enter a brief description for the rule.
6. Select the required access for other users.
7. Click Continue.
The version definition page is displayed.
8. Type the name of the version for the rule.
9. Select the effective start and end dates using the date picker. Alternatively, you can type them in the space provided.

Important: Each version must have a unique date range, as compared to all other versions for the same rule.

Note: If not specified otherwise in the related profile option, the defaults for the effective start and end dates are January 1, 1900 and December 31, 2500.

10. Specify any other properties or options that may apply for the version that you are creating.

11. Click Finish.

Please refer to *Overview of Common Rule Management Tasks*, page 3-1 for more information.

Viewing and Updating Rules

You can view existing rules and rule versions, and you can update certain properties for rules and versions.

Note the following considerations with regard to updating rules and versions:

- If you update a rule that has been approved, you must resubmit the rule for approval before it can be run.
- Rules that have been run are process locked, so you cannot update a rule that has been run except by removing the rule execution results and process lock. However, you can create a new (duplicate) version of a rule that has been run and edit then edit the new version without having to remove previous results.

Procedure:

1. Navigate to the home page, page 3-2 of the rule you want to update.
2. Search for a rule. For further information, see *Searching for Rules*, page 3-5.
3. Click Update corresponding to the rule or version that you want to update if you are familiar with the rule or version details and would like to update the rule or version directly. Alternatively, click on the rule or version to view details and then click Update on the View page.

Procedure to Update a Rule

1. Update the Name or Description.
2. Click Apply.

Duplicating Rules

You can duplicate rules and versions to avoid having to enter data multiple times. This saves time and effort and also reduces mistakes. You can duplicate only the version, or you can duplicate both the rule and the version.

When duplicating a version, the rule-related details cannot be updated. All existing versions for a rule are listed at the bottom of the duplicate page.

When you duplicate the version and the rule, a new rule is created with a copy of the version.

Procedure:

1. Navigate to the home page, page 3-2 of the rule or version you want to duplicate.
2. Search for a rule. For further information, see Searching for Rules, page 3-5.
3. Click Duplicate corresponding to the version of the rule that you want to duplicate.

Procedure to Duplicate a Version:

1. Select Version to create a new version in the same Rule.
2. Enter a unique name for the version.
3. (Optional) Enter a brief description for the version.
4. Select the effective start and end dates using the date picker. Alternatively, you can enter them in the space provided.

Caution: The new version's date range must not overlap with any of the existing version date ranges.

5. Click Finish.

Procedure to Duplicate a Rule and Version:

1. Select Rule and Version to create a new rule and a version.
2. Select the folder in which the rule will be stored.
3. Enter a name for the rule.
4. Enter a name for the version.
5. (Optional) Enter a description for the version.
6. Update the effective start and end dates using the date picker. Alternatively, you can type them in the space provided.
7. Click Finish.

Please refer to Overview of Common Rule Management Tasks, page 3-1 for more information.

Deleting Rules

You can delete rules that are no longer needed. To delete a rule, you delete all of the versions that are associated with that rule.

Caution: Once deleted, a rule cannot be retrieved.

Restrictions on deleting rules or versions are:

- You cannot delete rules or versions if you have only Read privileges. Only approvers or users with similar or higher system rights can delete rules or versions.
- A rule or version that has been approved can be deleted only by having a delete request approved by the approver.
- You cannot delete rules or versions if their approval is pending. Alternatively, you can delete the approval request and then the rule. However, this works only if you have sufficient privileges.
- You cannot delete versions associated with Locked Rules. A Locked Rule is one that has been already used in the production environment to generate final results.

Procedure:

1. Navigate to the home page, page 3-2 of the rule you want to delete.
2. Search for a rule. For further information, see Searching for Rules, page 3-5.
3. Click Delete corresponding to the rule or the version of the rule that you want to delete.

Please refer to Overview of Common Rule Management Tasks, page 3-1 for more information.

Extracting and Loading Rules

Note: Oracle Transfer Pricing in Release 12 does not yet support the extracting and loading rules functionality.

Enterprise Performance Foundation utilizes iSetup for extracting and loading rules in a folder within a single environment and for extracting and loading rules across multiple environments. For example, you can develop and test business rules in a testing environment and then load those rules to a production environment.

The Extract/Load features can also handle sub-objects. For example, the Mapping Rule

type "Retrieve Statistics" is comprised of a mapping rule and a statistic definition. The statistic definition also includes a Condition. When you extract or load the "Retrieve Statistics" rule, both the statistic definition and its condition will automatically be included in the process.

The Extract and Load options are available thru iSetup and let you initiate an Extract of a Folder's rules. Using the Load page you can load the extracted rules to the specified user's folder. You can extract, or load the following types of rules:

Dimension Member and Dimension Hierarchy Rules (facilitates moving metadata from one environment to another; available through the Enterprise Performance Foundation Administrator responsibility)

- Condition Rules (available through the iSetup Responsibility)
- Mapping Rules (available through the iSetup Responsibility)
- Transfer Pricing Rules (available through the Financial Transfer Pricing responsibility)

Prerequisites, Guidelines, and Considerations

In order to extract and load rules successfully, you must first ensure that the following requirements are met and that the systems are set up properly:

- Both the source and target environments should have the same set of languages installed. When you extract a set of rules, only one language will be extracted at a time.
- There should be a public database link between the source and target databases when loading rules between environments. The database link should be registered in Enterprise Performance Foundation (see Working with Registered Database Links, page 10-12 for more information.)
- Only rules with an Approval Status of: New, Approved, or Not Approved can be extracted. Rules with an Approval Status of Submitted for Approval or Submitted for Deletion cannot be extracted.
- When loading rules between environments, the username in the source environment must be the same as the username in the target environment.

Note: Potentially, any user in the target environment can load the rule and will have Read & Write access to it. However, if the rule was created in the source environment as Read Only, then the user who created the rule will be prevented from updating their rule unless that same user performs the load

- If the rule already exists as Read Only in the target environment, the load will only be successful if the loading username is the same as the username of the existing target rule.

- You should have folders with the same names in both environments.

Note: Access privileges to these folders should be set to **Read & Write**.

- The global value set combinations in both environments must match.
- The source and target environments should contain the same dimension members, groups, attributes, attribute Versions, and Hierarchies.

Note: Use the dimension and hierarchy migration tool to extract and load metadata from the source environment to the target environment.

- All hierarchies used in a rule should also be present in the target load environment. Loading an exported rule from a different environment with a hierarchy must have the same dimension members and hierarchy in place.
- (Oracle Profitability Manager users only) The target load environment's Activity and Cost Object composite dimension definitions must be identical to the definitions in the source extract environment.

For example, assume that the activity dimension is defined as: **CCC_ORG + Task +User Dimension_01** in the source environment. If the target load environment has Activity defined as: **CCC_ORG + Task**, the load will fail.

- When loading a rule that contains sub-objects in multiple folders, you must have **Read & Write** privileges for all of the folders used in the extracted Rule.

For example, a mapping rule in **Folder1** uses a condition that was created in **Folder2**. User1 extracts the rule. User2 must have **Read & Write** access to both folders in order to load the rule.

The Dimension Member and Dimension Hierarchy Migration and Loader Process

The migration and loader process is a simple two-step operation:

1. Login to Enterprise Performance Foundation Administrator in the target environment and migrate the dimension members and/or dimension hierarchies.
2. Login to Enterprise Performance Foundation Administrator in the target environment and load the dimension members and/or dimension hierarchies.

Step 1: Migrate the Dimension Members or Dimension Hierarchies

Follow these steps to migrate Dimension Members or Dimension Hierarchies

1. Login to Enterprise Performance Foundation Administrator in the target environment and navigate to the Process Management tab and select the Requests sub tab.
2. Migrate Dimension Members or Dimension Hierarchies

Note: A registered database link must be in place between the source and target environments.

To Migrate Dimension Members:

1. Schedule a Request by selecting **Dimension Member Migration** for the Program Name and click **Next**
2. Set Parameters: select a Source Database, select the Dimension to be migrated and click **Next**.
3. Enter Scheduling parameters (optional) and click **Next**.
4. Enter Recipient Notifications (optional) and click **Next**.
5. Enter Printing styles (optional) and click **Next**.
6. Review parameters and click **Submit**.

To Migrate Dimension Hierarchies:

1. Schedule a Request by selecting 'Dimension Hierarchy Migration' for the Program Name and click **Next**.
 2. Set Parameters: select a Database Link, select a Dimension, enter a Hierarchy from the source environment, enter the Hierarchy Version (optional), enter Dimension Name (optional) and click **Next**.
 3. Enter Scheduling parameters (optional) and click **Next**.
 4. Enter Recipient Notifications (optional) and click **Next**.
 5. Enter Printing styles (optional) and click **Next**.
 6. Review parameters and click **Submit**.
3. Navigate to the Process Management tab and select the View Requests sub tab to monitor the progress of the migration program.

Note: Clicking on the Details hyperlink of the request will display a page where you can view the log generated by the migration.

Step 2: Load the Dimension Members or Dimension Hierarchies

Follow these steps to load the Dimension Members or Dimension Hierarchies

1. Login to Enterprise Performance Foundation Administrator in the target environment and navigate to the Process Management tab and select the Requests sub tab.
2. Load Dimension Members or Dimension Hierarchies

Note: A registered database link must be in place between the source and target environments.

To Load Dimension Members:

1. Schedule a Request by selecting 'Dimension Member Loader' for the Program Name and click **Next**.
2. Set Parameters: select Execution Mode, select the Dimension to be loaded and click **Next**.
3. Enter Scheduling parameters (optional) and click **Next**.
4. Enter Recipient Notifications (optional) and click **Next**.
5. Enter Printing styles (optional) and click **Next**.
6. Review parameters and click **Submit**.

To Load Dimension Hierarchies:

1. Schedule a Request by selecting 'Dimension Hierarchy Loader' for the Program Name and click **Next**.
2. Set Parameters: select the **Hierarchy Loader** Rule Name, select the Execution Mode, select the Dimension, select the Hierarchy Name, select the Hierarchy Definition Name and click **Next**.
3. Enter Scheduling parameters (optional) and click **Next**.
4. Enter Recipient Notifications (optional) and click **Next**.
5. Enter Printing styles (optional) and click **Next**.

6. Review parameters and click **Submit**.
3. Navigate to the Process Management tab and select the View Requests sub tab to monitor the progress of the loader program.

Note: Clicking the Details hyperlink of the request will take you to a page where you can view the log generated by the loader.

The iSetup Extract and Load Process

The extract and load process is a simple two-step operation:

1. Login to iSetup in the source or target environment; extract the rules in a folder by selecting or creating a Selection Set.
2. Login to iSetup in the source or target environment; select the extract file and load.

Step 1: Extract the rules in a folder

Note: If Workflow is enabled, only versions with an Approval Status of: New, Approved, or Not Approved can be extracted.

Use the following procedure to extract rules in a folder

1. Login to iSetup and navigate to the Migrations tab and select the Selection Sets sub tab.
2. Create or select an existing Selection Set.

Note: A Selection Set contains the rules that the extract program will use to pull the types of objects in the selected environment.

To create a Selection Set

1. : click **Create** and select the template to pick which objects will be extracted:
Enterprise Performance Foundation Conditions
Profitability Manager Mapping Rules
Profitability Manager Statistics Rules
2. Click **Continue** and enter a name for the Selection Set and choose the Source Instance that the Selection Set will extract the objects from.

Note: A Filter can be applied to narrow the scope of Folders that the extract will draw from.

3. Click **Finish** to create the Selection Set.
3. Click **Extract** to navigate to the Set Parameters page and enter a name for the extract process to be executed.

Note: Users have the option to change the Selection Set and Source Instance; however, changing the Selection Set and/or Source Instance may invalidate the filters set at the time of the creation of the Selection Set.

4. Click **Continue** to navigate to the Schedule page and enter execution details.
5. Click **Finish** to submit the extract request. You will be returned to the Extracts page where you can view the progress of the extract.

Note: Clicking on the Name hyperlink of the extract will take you to a page where you can view the log generated by the extract.

Note: An extract will generate a .zip file that can only be used in a Load.

Step 2: Load the folder rules to an environment

Note: Loading rules to an environment different from the Source environment will require setting up a database link between two environments. See Working with Registered Database Links for more information.

Follow this procedure to load the folder rules to an environmentL

1. Login to iSetup and navigate to the Migrations tab and select the Extracts sub tab.
2. Search for an active Extract and select.

Note: An Active Extract is an extract that has a Normal Status indicator.

3. Click **Load** to navigate to the Set Parameters page and enter a name for the load process to be executed.

Note: Users have the option to change the extract load file by updating the Data Source field. Users also have the option to select the target environment for the load. The Target Instance will only display valid database links.

4. Click **Next** to view the Set Objects Details page. This is an informational page for the objects that will be created/updated in the target environment.
5. Click **Next** to navigate to the Schedule page and enter execution details.
6. Click **Finish** to submit the load request. You will be taken to the Loads page where you can view the progress of the load.
7. Clicking on the Name hyperlink of the load will display a page where you can view the log generated by the load.

iSetup Load Rule Behavior

If the rules of a folder have been loaded successfully, subsequent loads of the rules will either overwrite the rules in the folder if they exist for the same effective date range or add itself to the rule when the effective date ranges are different.

An extracted rule cannot be loaded into a rule where it will cross the effective date ranges of any two existing versions within the rule.

Furthermore, an extracted rule cannot be loaded into a rule where it will overwrite the effective date range of a rule that has been used in a calculation (a locked rule).

- Loads of rules to a different environment follow the same process and requirements as those for an extracted set of rules. You should pay special attention to the following requirements:
- The username in the target environment should be the same as the username in the source environment (Highly recommended, though not required; it is especially important in the case of rules with Read Only access).
- The Folder name of the Target environment must be the same as the Folder name of the Source environment.
- The user must have Read & Write access to the folder of the target environment.
- The dimensions and dimension members of the target environment must be the same as those in the source environment.

- The hierarchies of the target environment must be the same as those in the source environment.

Extracts or Loads of rules with sub-objects have the following distinctions:

- When loading, the user must have Read & Write access to the folder that contains any sub-objects of a rule.
- If several rules share the same sub-objects, a load of the extract will only create one copy of the sub-object.
- If several versioned sub-objects fall within the effective date range of the selected parent rule's version, they will be extracted.
- In general, if an error is encountered during extract, or load of a parent or sub-object, the extract or load will fail for both the parent and all its sub-objects, and the system will write appropriate error messages. This would include the case where no versioned sub-objects fall within the effective date range of the parent version.
- If the sub-object already exists for the same effective date range, the load of the sub-object will update the sub-object and prefix the name with "Import".
- If the sub-object already exists in the target environment, and the load effective date of the sub-object is different than the original, loading of the sub-object will adjust the effective dates accordingly.

For example, a rule version in the Test environment is Extracted and then Loaded into the Production environment.

- **Original Rule1 exists in the target Production environment and has one version ("Version1") for 1-Feb-2006 – 28-Feb-2006.**

Load rule version (also named "Version1") with an effective date range
1-Jan-2006 – 28-Feb-2006

The diagram consists of a legend and a grid. The legend shows a red hatched box for 'The effective date range of an existing rule version' and a blue hatched box for 'The effective date range of a rule version being imported'. The grid has two rows: 'Test' and 'Production', and columns for months from Dec to Oct. In the 'Test' row, the blue hatched box covers Jan and Feb. In the 'Production' row, the red hatched box covers Feb.

Test		The effective date range of a rule version being imported									
	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production			The effective date range of an existing rule version								

Result: The system will overwrite the target version, creating a version named "(Import) Version1" with the effective date range 1-Jan-2006 – 28-Feb-2006.

Result:

	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production											

- **Original Rule1 exists in the target environment and has one version ("Version1") for 1-Dec-2005 – 28-Feb-2006.**

Load rule version (also named "Version1") with an effective date range 1-Jan-2006 – 28-Feb-2006.

 The effective date range of an existing rule version
 The effective date range of a rule version being imported

Test	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production											

Result: The system will modify the effective date range of the existing target version, changing it to 1-Dec-2005 – 31-Dec-2005 and will write the loaded version (using the name "(Import) Version1"), with the effective date range 1-Jan-2006 – 28-Feb-2006.

Result:

	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production											

- **Original Rule1 exists in the target environment and has one version ("Version1") for 1-Dec-2005 – 31-Mar-2006.**

Load rule version (also named "Version1") is 1-Jan-2006 – 28-Feb-2006.



 The effective date range of an existing rule version
 The effective date range of a rule version being imported

Test	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production											

Result: The system will modify the effective date range of the existing target version, changing it to 1-Dec-2005 – 31-Dec-2005, it will write the loaded version (using the name "(Import) Version1") with the loaded effective date range of 1-Jan-2006 – 28-Feb-2006, and it will copy the original target version, naming it "(Copy) Version1" and setting the effective date range = 1-Mar-2006 – 31-Mar-2006.

Result:

	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct
Production											

- If the sub-object already exists, the extract of the sub-object cannot be loaded where it will cross the effective date ranges of any two existing versions within the rule.
- If the sub-object already exists, the extract of the parent and all sub-objects will fail and log an error message if any load requirements would be violated for the parent or sub-object. For example, if the parent or any sub-object would overwrite the effective date range of a version that has been used in a calculation (a locked version), the system will trigger an error

Working with Rule Approval Status

This chapter discusses the rule approval process and the procedure for managing approved rules.

This chapter covers the following topics:

- About Rule Approval and Production Data Sets
- The Rule Approval Process
- The Rule Deletion Process

About Rule Approval and Production Data Sets

Oracle Transfer Pricing (FTP) uses Oracle Approvals Management (AME) and Oracle Workflow to manage the approval status of rules.

Note: The approval process actually applies to versions of rules, as opposed to rules themselves. For simplicity's sake, however, the term "rule" is used in this chapter to refer to versions of rules.

A rule can write results to a production data set only if the status of the rule is Approved. Rules that have not been approved can write results only to non-production data sets. Note that a production data set is defined as one for which the appropriate attribute has been set on the data set member.

If you want to use production data sets, you must set up an approval hierarchy through Oracle Approvals Management (AME), where you must specify **FEM Approvals** as the transaction type. After the approval hierarchy has been defined, Oracle Transfer Pricing uses Oracle Workflow to route the approval requests to the appropriate approvers.

If a rule has been run and there are saved results that have been generated by the rule, the rule definition is locked, and you cannot change or delete the rule definition. You can only change or delete a rule definition if there are no existing results for that rule.

For further information about Oracle Approvals Management and Oracle Workflow, see the following:

- *Oracle Approvals Management Implementation Guide*
- *Oracle Workflow User's Guide*

For more information about the processes for rule approval and deletion, see the following topics:

- The Rule Approval Process, page 4-2
- The Rule Deletion Process, page 4-2

The Rule Approval Process

When you first create a rule, the status for the rule is New. The rule must be approved before you can use it to write results to a production data set (you can, however, test an unapproved rule against a non-production data set).

When you submit a rule for approval, the status becomes Submit Approval. The approver can then either approve or reject the rule. Note that you cannot run a rule for which the status is Submit Approval against either production or non-production data sets.

If the approver approves the rule, the rule status changes to Approved, and you can now use the rule to write results to production data sets.

If the approver rejects the rule, the rule status changes to Not Approved. If there are no existing results for that rule definition, you can make changes to the rule definition and resubmit the rule for approval.

If you try to change the definition for an approved rule for which there are no existing results, a backup copy of the approved rule definition is created. The status of this backup copy of the definition is Not Approved. You can then revert to this backup copy of the rule definition, edit it as desired, and then submit the revised definition for approval.

The Rule Deletion Process

To delete a rule definition for which the status is Approved, you must obtain approval to delete the rule before it can be deleted. Therefore, to delete an approved rule, you must submit the rule definition for deletion approval.

Click the **Delete** icon for a rule on the rule's home page submit the rule definition for deletion approval. This action launches the approval process and routes the request for deletion to the appropriate approver.

When you submit a rule for deletion approval, the definition status for the rule becomes Submit Delete. Note that you cannot run a rule for which the status is Submit Delete against either production or non-production data sets.

If the approver approves the deletion of an approved rule, the rule is automatically

deleted. If the approver rejects the deletion, the rule definition status is reset to Approved.

Rules with a status of New — that is, rules that have not been submitted through the approval process — do not require approval for deletion. As long as there are no results in a non-production data set as a result of running such a rule, you can simply delete the rule.

Managing Currency Rates

This chapter discusses the process for managing currency rate information, including creating daily, historical, and cross rates. The chapter also describes how to upload daily rates and upload and download historical rates from spreadsheet-based applications to Oracle Transfer Pricing.

This chapter covers the following topics:

- Overview of Currency Rates Management
- Overview of Multi-Currency Accounting
- Currencies
- Conversion Rates
- Entering Daily Rates
- Loading Daily Rates Automatically
- Entering Historical Rates
- Revaluing Balances
- Translating Balances
- Currency Rates Manager

Overview of Currency Rates Management

Financial institutions, such as banks, usually transact in more than one currency and this necessitates multi-currency accounting, which, in turn, requires currency rates management. See: Overview of Multi-Currency Accounting, page 5-3.

Oracle Transfer Pricing provides you with the currency rates management functionality through the Currency Rates Manager, a part of the Oracle General Ledger (GL) application. See:

- Currency Rates Manager, page 5-57.

- Using the Daily Rates Spreadsheet Interface, page 5-61.
- Using the Historical Rates Spreadsheet Interface, page 5-62.

Using Currency Rates Manager, you can input cross-currency exchange rates for currencies that have been enabled in the application.

Note: To enable or disable currencies in the application, you require an appropriate General Ledger responsibility, such as General Ledger Super User. You must decide during setup itself which currencies should be active in the application and enable them. See: Defining Currencies, page 5-9.

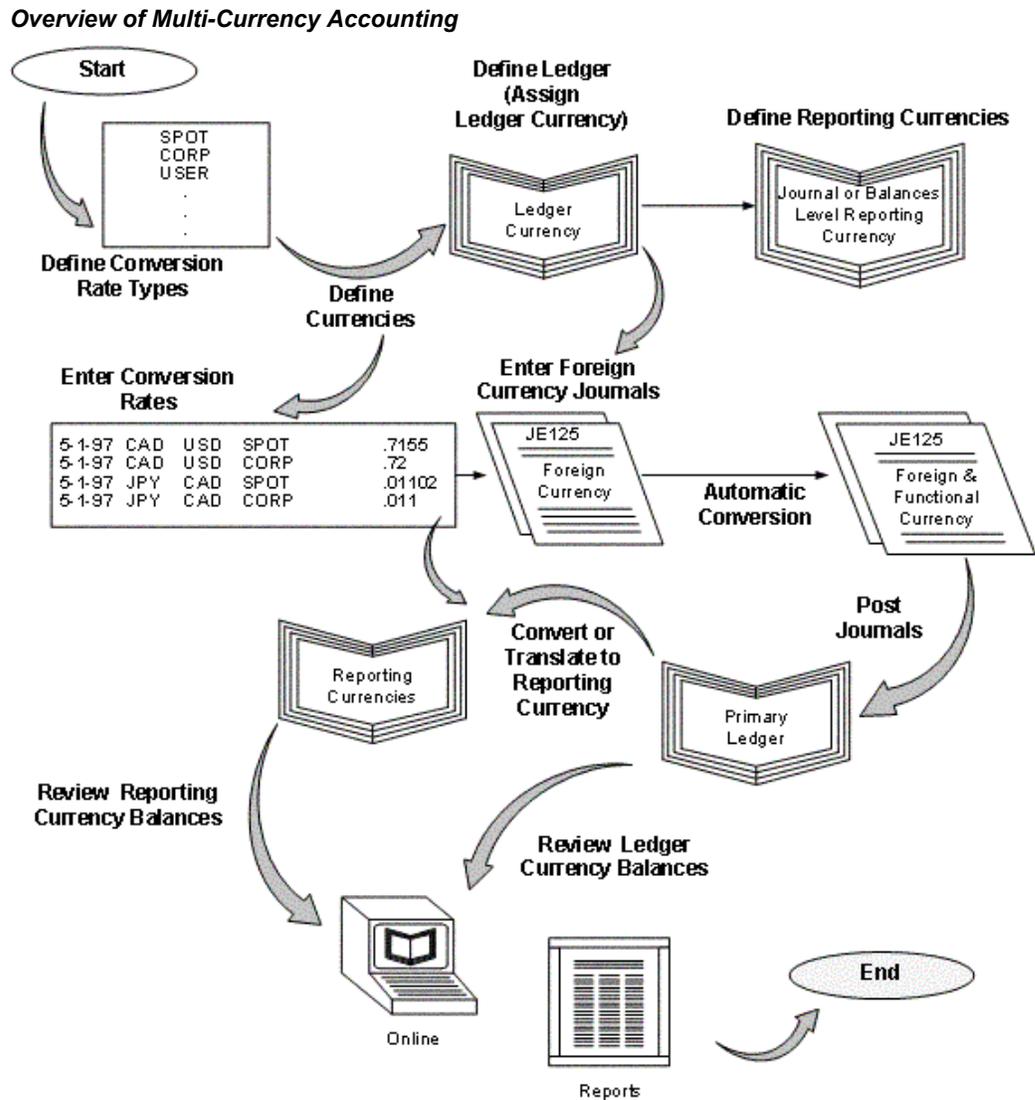
Oracle Transfer Pricing makes use of the cross-currency exchange rates during the charge/credit calculation and migration process. The cross-currency exchange rates functionality enables you to store data in the local or entered currency but convert it to a reporting currency during charge/credit migration.

Related Topics

Transfer Pricing Process Rule and Migration Options, page 2-43

Standard Navigation Paths, page A-1

Overview of Multi-Currency Accounting



Overview

Oracle General Ledger provides full multi-currency functionality to meet the needs of global companies. This chapter introduces multi-currency concepts in accordance with the United States Statement of Financial Accounting Standards 52 (SFAS #52) and International Accounting Standards 21 (IAS 21) requirements as they apply to General Ledger.

Determining the Functional and Ledger Currency

Your organization's ledger currency as discussed in SFAS #52 and IAS 21 can be different from the General Ledger ledger currency. For example, you may choose Japanese Yen (JPY) for your ledger currency when your ledger currency for the accounting purposes of your integrated business group is actually US Dollars (USD). The determination of the ledger currency is based on a number of factors, discussed in SFAS #52 and IAS 21. The ledger currency represents the base currency that Oracle General Ledger will maintain for your ledger.

Translation vs. Remeasurement

General Ledger performs translation in compliance with multiple national accounting standards, in particular SFAS #52 and IAS 21. General Ledger can perform two types of translation stipulated in SFAS #52 and IAS 21:

1. Translation or Equity Translation Method
2. Remeasurement or Temporal Method Translation

The translation method you use depends on your ledger currency:

1. If the ledger currency is different from the ledger currency, books of record must be remeasured into the ledger currency before being translated into the reporting currency. If the ledger currency is the reporting currency, remeasurement eliminates the need for translation.
2. If the ledger currency is the same as the ledger currency, books of record can be directly translated into the reporting currency without remeasurement.

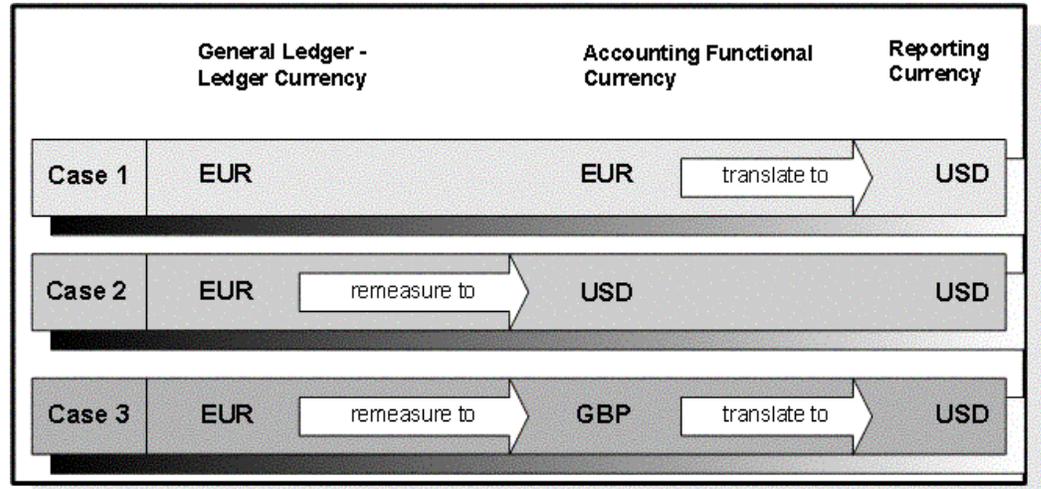
The example and table, below, illustrates when translation and remeasurement are required.

Consider U.S. company A has a wholly owned subsidiary, company B, that operates in Europe. The reporting currency is USD. The translation–remeasurement possibilities for company B are listed horizontally by case examples in the following diagram:

In Case 1, Company B's ledger currency and ledger currency are the same. Company B's ledger currency is not the same as the parent's functional or reporting currency. However, this usually arises when the subsidiary is not integral to the parent's business. In order to meet the reporting requirements of the parent, Company B has to perform translation from EUR to USD.

In Case 2, Company B's ledger currency is different from its ledger currency. However, Company B's ledger currency is the same as the parent's functional or reporting currency. This usually arises when the subsidiary is integral to the parent's business and cannot be sold without a severe impact to the parent. Company B remeasures its accounts from EUR to USD, which is also the reporting currency. In this case, remeasurement into the reporting currency eliminates the need for translation.

In Case 3, Company B's ledger currency is GBP. Company B's ledger currency is neither its ledger currency nor the reporting currency. This is a rare case and could arise if the subsidiary is a holding company for operations in England. In this case, both remeasurement and translation are required. Company B first remeasures its accounts from EUR to GBP, then translates the accounts from GBP to USD.



For an in depth discussion on how to implement Oracle functionality to evaluate the results of overseas operations in accordance with US and International accounting standards, please refer to the Parent Currency View of Overseas Operations whitepaper on Metalink.

Concepts

Throughout this chapter, we discuss the following concepts relating to multi-currency accounting in Oracle General Ledger.

Processes

There are three key processes in Oracle General Ledger to address multi-currency requirements::

- **Conversion:** refers to foreign currency transactions that are immediately converted at the time of entry to the ledger's currency in which the transaction takes place.
- **Revaluation:** adjusts asset or liability accounts that may be materially understated or overstated due to a significant fluctuation in the exchange rate between the time the transaction was entered and the time revaluation takes place.
- **Translation:** restates an entire ledger or balances for a company from the ledger currency to another currency. The cumulative translation adjustment is typically recorded as part of equity.

Remeasurement: restates an entire ledger or balances for a company from the ledger

currency to another currency. For non-monetary items, remeasurement uses historical rates. The cumulative translation adjustment is typically recorded as part of profit or loss.

To use multi-currency accounting, you must first define Currencies and Conversion Rate Types. Currency processes perform revaluation, translation, and remeasurement using daily or historical rates that you enter. Daily rates can be entered manually, using a spreadsheet interface or loaded in the GL_DAILY_RATES_INTERFACE table using SQL instructions. For more information on spreadsheet interface, see: Currency Rates Manager, page 5-57.

Setting Up Multi-Currency Accounting in Oracle General Ledger

To implement multi-currency accounting in General Ledger, follow the recommended setup steps listed below.

To set up multi-currency accounting:

1. Define the conversion rate types you want to use to maintain daily exchange rates and to enter foreign currency journals. General Ledger comes with three predefined conversion rate types: Spot, Corporate, and User. See: Defining Conversion Rate Types, page 5-11.
2. Define and enable the currencies you want to use. General Ledger predefines all ISO currencies, but you can define as many additional currencies as you need. See: Defining Currencies, page 5-9.
3. Assign a ledger currency to your ledger. General Ledger records all transactions and maintains all of your account balances in the ledger currency. See: Defining Ledgers, *Oracle General Ledger Implementation Guide*.
4. Define a Cumulative Translation Adjustment account for your ledger. Set the account type of your Cumulative Translation Adjustment account to:
 - **Owner's Equity:** to create a translation adjustment on your balance sheet if you perform translation.
 - **Revenue or Expense:** to create a translation gain/loss on your income statement if you perform remeasurement.

General Ledger automatically posts any net adjustments resulting from currency translation to the Cumulative Translation Adjustment account in accordance with SFAS #52 and IAS 21.

Note: General Ledger conforms to multiple national accounting standards, including SFAS #52 (U.S.)--with regard to the translation, revaluation, and reporting of foreign

currency-denominated balances.

5. Define an account to use to record unrealized gains and losses that arise when you revalue account balances that are denominated in a foreign currency. See: *Defining Accounts, Oracle General Ledger Implementation Guide* and *Revaluing Balances*, page 5-27.
6. Enter the daily rates you will need. Typically, you will enter rates for foreign-entered journals, revaluation, translation, and remeasurement. See *Entering Daily Rates*, page 5-13.

If you do not want to predefine daily rates, you can use the conversion rate type User to enter daily rates at the time you enter journals.

Note: If you have average balance processing enabled in your ledger, you must define a daily rate on or before the first day of the first year for which you want to translate balances.

If you use reporting currencies or secondary ledgers (journal or subledger level), daily rates are used to convert journals from the source ledger to the reporting currencies or secondary ledgers.

7. Assign the rate types for your ledger to be used as your period-average and period-end rates for running foreign currency revaluation or translation.
8. Enter historical rates or amounts to translate selected balances in accordance with applicable accounting standards. General Ledger also uses historical rates and amounts to remeasure selected account balances for companies in highly inflationary economies. See: *Entering Historical Rates*, page 5-23.
9. (Optional) Enable secondary segment tracking for your ledger in Accounting Setup Manager. You can optionally identify a secondary tracking segment to produce more detail for retained earnings, unrealized gain and loss, and cumulative translation adjustment accounts by using a primary balancing and secondary tracking segment. You must first assign the Secondary Tracking Segment Qualifier to a segment in your accounting flexfield. You cannot select the primary balancing segment, intercompany segment, or natural account segment as the Secondary Tracking Segment Qualifier. See: *Secondary Tracking Segment, Oracle General Ledger User's Guide*.

Using Multi-Currency Accounting in Oracle General Ledger

To use multi-currency accounting in General Ledger, review the steps detailed below.

To work with multiple currencies in General Ledger: To work with multiple currencies in General Ledger:

1. Update your daily conversion rates regularly.
2. Enter or import foreign currency journals. If you use the conversion rate type User, enter the currency conversion rate when you enter journals. See: *Entering Entered Currency Journals, Oracle General Ledger User's Guide*.
3. Post your foreign currency journal entries to an open period. General Ledger stores the foreign currency amount associated with each journal line, in addition to the converted ledger currency equivalent. See: *Posting Journal Batches, Oracle General Ledger User's Guide*.
4. Revalue foreign currency-denominated accounts. General Ledger creates journal entries to adjust the ledger currency balances for exchange rate fluctuations, in accordance with SFAS #52 and IAS 21. See: *Revaluing Balances, page 5-27*.
5. Post the revaluation journal batch to adjust your unrealized gain/loss account for exchange rate fluctuations. See: *Posting Journals, Oracle General Ledger User's Guide*.
6. Translate account balances before consolidating ledgers with different ledger currencies, or translate to report account balances in another currency. You can translate actual or budget balances. See: *Translating Balances, page 5-36*.

Note: If you use Reporting Currencies, you can report account balances in another currency directly from your reporting currency (journal or subledger transaction level). See: *Overview of Reporting Currencies, Oracle General Ledger User's Guide*.

7. Review entered and converted foreign currency balances online using the Account Inquiry window. You can also review translated amounts online using the Account Inquiry window. See: *Performing an Account Inquiry, Oracle General Ledger User's Guide*.

Note: You must have previously translated your account balances to the foreign currency before you can perform the translated account balance inquiry.

8. Run Trial Balance reports. Use the:
 - Trial Balance, Detail, Additional Segment Detail or Translation Trial Balances to view translated account balances after you run translation.
 - Trial Balance, Detail, or Additional Segment Detail Trial Balances to view balances entered in a foreign currency.

- Entered Currency General Ledger Report to reconcile revaluation journals after you run revaluation.
9. See: Running Standard Reports and Listings, *Oracle General Ledger User's Guide*.
 10. Produce foreign currency financial statements. Use the Financial Statement Generator to build custom reports to report on actual and budget translated account balances, as well as amounts entered in foreign currency. See: Overview of the Financial Statement Generator, *Oracle General Ledger User's Guide*.

Related Topics

Defining Currencies, page 5-9

Entering Daily Rates, page 5-13

Loading Daily Rates Automatically, page 5-18

Entering Entered Currency Journals, *Oracle General Ledger User's Guide*

Defining Conversion Rate Types, page 5-11

Translating Balances, page 5-36

Revaluing Balances, page 5-27

Overview of Multi-Currency Accounting, page 5-3

Currency Rates Manager, page 5-57

Setting General Ledger Profile Options, *Oracle General Ledger Reference Guide*

Overview of Reporting Currencies, *Oracle General Ledger User's Guide*

Currencies

Defining Currencies

Use the Currencies window to define non-ISO (International Standards Organization) currencies, and to enable/disable currencies. Oracle Applications has predefined all currencies specified in ISO standard #4217.

To use a currency other than U.S. Dollars (USD), you must enable the currency. U.S. Dollars (USD) is the only currency that is enabled initially.

Currencies Window

Code	Name	Description	Issuing Territory	Symbol	Precision
ALL	Lek	Lek	Albania		2
					2

To define a new currency:

1. Navigate to the Currencies window.
2. Enter a unique Code to represent your currency.

Note: You cannot change a currency code after you enable the currency, even if you later disable that currency.
3. Enter the Name and Description of the currency.
4. (Optional) Select the name of the Issuing Territory. Oracle Applications has predefined the names of countries (per ISO Standard #3166) that issue standard currencies.
5. Enter the Symbol for your currency.

Note: Some Oracle Applications use currency symbols when displaying amounts. Others, like General Ledger, do not.
6. Enter the Precision of the currency to designate the number of digits to the right of the decimal point used in regular currency transactions.
7. Enter the Extended Precision to designate the number of digits to the right of the decimal point used in calculations for this currency. The extended precision must be greater than or equal to the standard precision.

Note: Some Oracle Applications use the extended precision. Others, like General Ledger, do not.

8. Enter the Minimum Accountable Unit to designate the smallest denomination used in this currency. Note that this might not correspond to the precision.
9. (Optional) Enter Effective Dates for your currency. You can only enter transactions denominated in this currency for dates within the range. If you don't enter a start date, the currency is valid immediately. If you don't enter an end date, the currency is valid indefinitely.
10. Enable your currency.
11. Save your work.

To enable or disable a currency:

1. Navigate to the Currencies window.
2. Query the Code or Name of the currency that you want to enable or disable.
3. Mark the Enabled check box to indicate that the currency can be used to enter transactions and record balances. Clear the check box to indicate that the currency cannot be used.
4. Save your work.

Related Topics

Defining Currencies, page 5-9

Overview of Multi-Currency Accounting, page 5-3

Conversion Rates

Defining Conversion Rate Types

Use conversion rate types to automatically assign a rate when you:

1. convert foreign currency journal amounts to your ledger currency equivalents
2. run Revaluation
3. run Translation or Remeasurement

You enter daily conversion rates for specific combinations of foreign currency, date, and

conversion rate type. When you enter a foreign currency journal, General Ledger automatically displays the predefined exchange rate based on the currency, rate type (unless you are using the User rate type), and conversion date you enter. When you have a User rate type, you enter the rate directly when you enter a foreign currency journal.

Note: If you want to enter different daily rates for the same combination of from-currency, to-currency, and conversion date, you must define separate conversion rate types.

General Ledger provides the following predefined daily conversion rate types:

Spot:An exchange rate which you enter to perform conversion based on the rate on a specific date. It applies to the immediate delivery of a currency.

Corporate: An exchange rate you define to standardize rates for your company. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.

User:An exchange rate you specify when you enter a foreign currency journal entry.

You can use these predefined rate types to enter exchange rates, or you can define additional conversion rate types. After defining a conversion rate type, enter daily rates using that rate type.

To define a new conversion rate type:

1. Navigate to the Conversion Rate Types window.
2. Enter a Name and Description for the new conversion rate type.
3. (Optional) Select the Enable Security checkbox to apply Definition Access Set security to your conversion rate type.

Definition Access Sets are an optional security feature that allows you to control access to your General Ledger definitions. For example, you can prevent certain users from viewing, making changes, or using your conversion rate type.

If you do not enable security, all users will be able to use, view, and modify your conversion rate type.

If the Assign Access function is available for your responsibility, the Assign Access button will be enabled once you check the Enable Security check box. Choose the Assign Access button to assign the definition to one or more Definition Access Sets with the desired privileges.

For more information, see the Definition Access Set for Conversion Rate Type table in the Definition Access Set Security section of this chapter. It explains the Use, View, and Modify privileges to the Conversion Rate Types in the Daily Rates window. Also see Definition Access Sets, *Oracle General Ledger User's Guide*.

If the Assign Access function has been excluded from your responsibility, you will not be able to view the Assign Access button in the Conversion Rate Types window. You can still secure the Conversion Rate Type by checking the Enable Security check box, but only Definition Access Sets that are AutoAssigned will be automatically assigned to this Conversion Rate Type. For more information on Function Security, see your System Administrator.

4. Save your work.

Related Topics

Entering Entered Currency Journals, *Oracle General Ledger User's Guide*

Entering Daily Rates, page 5-13

Defining Currencies, page 5-9

Overview of Multi-Currency Accounting, page 5-3

Entering Daily Rates

General Ledger uses daily rates to perform foreign currency journal Conversions, revaluation, and translation/remeasurement. You can maintain daily conversion rates between any two currencies that you are enabled in your applications instance. In addition, you can enter a single exchange rate for a range of dates in the Enter Rates By Date Range window. The date range can span multiple days or periods.

If you use Reporting Currencies (journal or subledger level), your daily rates are used to convert your ledger's journals to the appropriate reporting currencies during posting. Your daily rates must be defined before you post journals in your ledger.

Entering Foreign Currency Journals

If you specify a foreign currency, conversion date, and conversion rate type when entering journals, General Ledger will automatically display the daily rate you defined to convert the foreign currency to your ledger currency, for the specified date and rate type. General Ledger calculates functional debit and credit equivalents by multiplying the debits and credits entered in a foreign currency by the retrieved daily rate.

See: Entering Entered Currency Journals, .

Using Period-End and Period-Average Rates in Translation

According to SFAS #52 and IAS 21, the period-end rate represents the rate at the balance sheet date and the period-average rate represents the average exchange rate. General Ledger uses Period - average and period-end rates when you translate your actual and budget account balances.

Typically, you use period-average rates to translate income statement accounts and

period-end rates to translate balance sheet accounts. The default period-average and period-end rate types must be assigned when you create the ledger.

Oracle General Ledger enables you to assign a conversion rate type for your period-end and period-average rates to comply with the accounting standards. You can assign any conversion rate type as your period-average and period-end rates for the ledger. For example, you can assign the predefined rate type Spot to be used as your period-average rates and the predefined rate type Corporate to be used as your period-end rates. These rate types are used in translation of actual account balances.

For budget account balances, you can specify the period-end and period-average rate types when you submit the translation.

To define your period-end and period-average rates:

1. Create a new conversion rate type or use a predefined rate type in the Conversion Rate Type window.
2. Enter rates for the conversion rate type in the Daily Rates window.
If your conversion rate type is assigned to a definition access set, you must have Modify and View privileges to enter rates for the conversion rate type.
3. Assign the conversion rate type to the period-end and period-average rates for your ledger using the Accounting Setup Manager.

Note: If you change a period-average or period-end rate for a period in which you have already run translation, you must retranslate your account balances for that period.

Defining Conversion Rate Types,

Entering Daily Rates,

Average Daily Balances

If you have average balance processing enabled in your ledger, you must define a daily rate on or before the first day of the first year for which you want to translate balances.

Definition Access Set Security

General Ledger maintains one set of daily rates for all ledgers within an Applications instance. You can use Definition Access Sets to control access to your daily rates by securing the conversion rate types. For example, you can prevent certain users from viewing, updating, or creating rates using your conversion rate type.

The following table explains what Use, View, and Modify privileges mean for the Conversion Rate Type definition in the Conversion Rate Type and Daily Rates windows.

Definition Access Set for Conversion Rate Type

Window	Use Privilege	View Only Privilege	Modify Privilege (with View Privilege)
Conversion Rate Type	Use or assign the rate type when entering journals, defining MassAllocations, running Revaluations, assigning period-average and period-end rate types, etc.	View Rate Type	Update conversion rate type name or description
Daily Rates	Not Applicable	View daily rates associated with the conversion rate type	Create, update, and delete daily rates associated with the conversion rate type

Prerequisites

1. Define and enable your currencies.
2. Define your conversion rate types.

Note: If you want to enter different daily rates for the same combination of from - currency, to - currency, and conversion date, you must define separate conversion rate types.

See: Defining Conversion Rate Types, .

Have your system administrator set the profile option Daily Rates Window: Enforce Inverse Relationship During Entry.

To enter a daily conversion rate:

You can use the Daily Rates window or Currency Rates Manager to enter: daily rates, daily rates by date range, and daily rates using a spreadsheet. See: Currency Rates Manager.

1. Navigate to the Daily Rates window.
2. Enter the From-Currency - the currency you want to convert from using the rates you enter. You can choose any enabled currency except STAT.
3. Enter the To - Currency - the currency to which you want to convert. If you enter

the same currency as your from - currency, you will receive an error.

4. Enter the Conversion Date and Type. When you use this date and rate type to enter journals, General Ledger automatically displays the rate you define here.

You cannot select the rate type of User. Enter User rate directly on the Enter Journal window when creating a foreign entered journal.

If your conversion rate type is assigned to a definition access set, you must have Modify and View privileges to the conversion rate type to create new rates.

5. Enter the conversion rate you want General Ledger to use to convert your from-currency amounts into your to-currency equivalents. General Ledger automatically calculates the inverse of the rate and displays it in the adjacent column.

If the profile option Daily Rates Window: Enforce Inverse Relationship During Entry is set to Yes, General Ledger ensures that the rates in both columns always have an inverse relationship. If either rate is changed, General Ledger automatically recalculates the other as the inverse of the changed rate.

If the profile option is set to No, General Ledger will not enforce the inverse relationship. You can change either of the rates independently.

- Enter a rate in the first column that converts your from-currency to your to-currency. This is the rate that you multiply your from-currency amount by to determine the to-currency equivalent. For example, to convert AUD to USD (Australian Dollars to U.S. Dollars), enter .7793 if the rate is .7793 U.S. dollars per Australian dollar.
- Enter a rate in the second column that converts your to-currency to your from-currency. This is the rate that you multiply your to-currency amount by to determine the from-currency equivalent. For example, to convert USD to AUD (U.S. Dollars to Australian Dollars), enter 1.2832 if the rate is 1.2832 Australian dollars per U.S. dollar.

Note: If you have the profile option Journals: Display Inverse Rate set to Yes, General Ledger will display inverse exchange rates in the Enter Journals and other windows. For example, assume that the profile option is set to Yes and your ledger currency is USD. If you enter the AUD to USD rate as .7793 in the Daily Rates window, General Ledger will display the inverse rate, or 1.2832, in the Enter Journals window when you create a foreign currency journal using AUD as the foreign currency.

To enter a single rate for a date range:

1. Navigate to the Daily Rates window.
2. Choose the Enter by Date Range button.
The Enter Rates By Date Range window appears.
3. Enter the From-Currency - the currency you want to convert from using the rates you enter. You can choose any enabled currency except STAT.
4. Enter the To - Currency - the currency to which you want to convert. If you enter the same currency as your from - currency, you will receive an error.
5. Enter From Date and To Date to span your desired date range.
6. Enter the Conversion Date and Type. When you use this date and rate type to enter journals, General Ledger automatically displays the rate you define here. If your conversion rate type is assigned to a definition access set, you must have Modify and View privileges to create new rates.
7. Enter the conversion rate you want General Ledger to use to convert your from-currency amounts into your to-currency equivalents. General Ledger automatically calculates the inverse of the rate and displays it in the adjacent column.

If the profile option Daily Rates Window: Enforce Inverse Relationship During Entry is set to Yes, General Ledger ensures that the rates in both columns always have an inverse relationship. If either rate is changed, General Ledger automatically recalculates the other as the inverse of the changed rate.

If the profile option is set to No, General Ledger will not enforce the inverse relationship. You can change either of the rates independently.

- Enter a rate in the first column that converts your from-currency to your to-currency. This is the rate that you multiply your from-currency amount by to determine the to-currency equivalent. For example, to convert AUD to USD (Australian Dollars to U.S. Dollars), enter .7793 if the rate is .7793 U.S. dollars per Australian dollar.
- Enter a rate in the second column that converts your to-currency to your from-currency. This is the rate that you multiply your to-currency amount by to determine the from-currency equivalent. For example, to convert USD to AUD (U.S. Dollars to Australian Dollars), enter 1.2832 if the rate is 1.2832 Australian dollars per U.S. dollar.

Note: If you have the profile option Journals: Display Inverse Rate set

to Yes, General Ledger will display inverse exchange rates in the Enter Journals and other windows. For example, assume that the profile option is set to Yes and your ledger currency is USD. If you enter the AUD to USD rate as .7793 in the Daily Rates window, General Ledger will display the inverse rate, or 1.2832, in the Enter Journals window when you create a foreign currency journal using AUD as the foreign currency.

Note: If you run a query on a single rate that contains a range of dates, your query results list each date within your specified date range as a single row.

Loading Daily Rates Automatically

General Ledger provides the `GL_DAILY_RATES_INTERFACE` table that you can use to automatically insert, update, or delete daily rates in the `GL_DAILY_RATES` table. General Ledger validates the rows in the interface table before making changes in the `GL_DAILY_RATES` table.

Warning: Always use the interface table to load your daily rates into General Ledger. Do not load rates directly into the `GL_DAILY_RATES` table, since this can corrupt your daily rates data.

When General Ledger processes the interface table, the system follows the behavior described below:

- If you specify a range of conversion dates, the system inserts, updates, or deletes one row in `GL_DAILY_RATES` for each date in your range. For example, if you specify:

```
From-currency:      JPY
To-currency:        USD
Conversion date range: 01-OCT-97 to 03-OCT-97
User conversion type: Spot
Conversion rate:    .0083
```

... and you are inserting new rates, General Ledger will insert three new rows into `GL_DAILY_RATES` with the following information:

```
JPY USD 01-OCT-97 Spot .0083
JPY USD 02-OCT-97 Spot .0083
JPY USD 03-OCT-97 Spot .0083
```

- General Ledger automatically inserts, updates, or deletes the corresponding inverse rates rows in `GL_DAILY_RATES`. Using the same example as above, General Ledger will insert three additional rows into `GL_DAILY_RATES` with the following information:

```

USD JPY 01-OCT-97 Spot 120.482
USD JPY 02-OCT-97 Spot 120.482
USD JPY 03-OCT-97 Spot 120.482

```

The GL_DAILY_RATES_INTERFACE Table

With the introduction of the Currency Rates Manager, the GL_DAILY_RATES_INTERFACE.AL trigger is disabled upon upgrade to this feature. To continue using the trigger logic, set the GL_CRM_UTILITIES_PKG.ENABKE_TRIGGER:=TRUE.

To enable cross rates logic either call the public Application Program Interface (API) GL_CRM_UTILITIES_PKG.DAILY_RATES_INPUT or run the Daily Rates Import and Calculation program from the Submit Request window.

See: Currency Rates Manager, page 5-57.

The columns in GL_DAILY_RATES_INTERFACE are described in the table below.

GL_DAILY_RATES_INTERFACE Table

Column Name	Null?	Type
FROM_CURRENCY	NOT NULL	VARCHAR2 (15)
TO_CURRENCY	NOT NULL	VARCHAR2 (15)
FROM_CONVERSION_DATE	NOT NULL	DATE
TO_CONVERSION_DATE	NOT NULL	DATE
USER_CONVERSION_TYPE	NOT NULL	VARCHAR2 (30)
CONVERSION_RATE	NOT NULL	NUMBER
MODE_FLAG	NOT NULL	VARCHAR2 (1)
INVERSE_CONVERSION_RATE		NUMBER
USER_ID		NUMBER (15)
ERROR_CODE		VARCHAR2 (30)

Column Name	Null?	Type
LAUNCH_RATE_CHANGE		VARCHAR2 (1)
CONTEXT		VARCHAR2 (150)
ATTRIBUTE1		VARCHAR2 (150)
ATTRIBUTE2		VARCHAR2 (150)
ATTRIBUTE3		VARCHAR2 (150)
ATTRIBUTE4		VARCHAR2 (150)
ATTRIBUTE5		VARCHAR2 (150)
ATTRIBUTE6		VARCHAR2 (150)
ATTRIBUTE7		VARCHAR2 (150)
ATTRIBUTE8		VARCHAR2 (150)
ATTRIBUTE9		VARCHAR2 (150)
ATTRIBUTE10		VARCHAR2 (150)
ATTRIBUTE11		VARCHAR2 (150)
ATTRIBUTE12		VARCHAR2 (150)
ATTRIBUTE13		VARCHAR2 (150)
ATTRIBUTE14		VARCHAR2 (150)
ATTRIBUTE15		VARCHAR2 (150)
USED_FOR_AB_TRANSLATI ON		VARCHAR2 (1)

Required and Conditionally Required Columns

The field descriptions below are based on the example below.

FROM_CURRENCY: The source currency applicable to the conversion rate. The amount denominated in the from-currency multiplied by the conversion rate gives the amount denominated in the to-currency.

TO_CURRENCY: The target currency applicable to the conversion rate.

FROM_CONVERSION_DATE: The starting date of the range of dates for which rows will be inserted into GL_DAILY_RATES. General Ledger will insert one row for each date in the range. Each date will have the same conversion rate you specify.

TO_CONVERSION_DATE: The ending date of the range of dates for which rows will be inserted into GL_DAILY_RATES.

Note: The range of dates specified by FROM_CONVERSION_DATE and TO_CONVERSION_DATE cannot exceed 366 days.

USER_CONVERSION_TYPE: The conversion type that users see displayed in the Daily Rates window. General Ledger automatically converts the user conversion type into the conversion type ID that is stored in the GL_DAILY_RATES table.

CONVERSION_RATE: The currency conversion rate. This is the rate by which the amount denominated in the from-currency is multiplied to arrive at the amount denominated in the to-currency.

Note: If the row you are entering in the interface table is to delete rates in GL_DAILY_RATES, enter a dummy CONVERSION_RATE.

MODE_FLAG: For each row, enter 'D' if you want to delete matching rows from the GL_DAILY_RATES table. Enter 'I' if you want to insert new rows.

Note: If you specify 'I' as the MODE_FLAG and the combination of from-currency, to-currency, conversion date, and user conversion type already exist in GL_DAILY_RATES, the existing rate will be updated with the new rate you specified in the interface table.

If you specify 'D' as the MODE_FLAG, General Ledger will also delete corresponding inverse rates rows in GL_DAILY_RATES.

Note: Any rows you enter in GL_DAILY_RATES_INTERFACE that fail validation will remain in the interface table and will not be moved to GL_DAILY_RATES. Also, the mode flag will change to X and the error code column will be populated. Use a SQL*Plus SELECT statement to check if any of the rows you loaded into the interface table failed validation.

You cannot reprocess rejected rows that remain in the interface table after failing validation. To process the correct data, you must first

delete the rejected rows from the interface table then enter the correct data as new rows in the table. The new data will be processed as usual.

Optional Columns

INVERSE_CONVERSION_RATE: The inverse of the conversion rate. This is the rate by which the amount denominated in the to-currency is multiplied to arrive at the amount denominated in the from-currency.

Note: If you do not provide this value, General Ledger will calculate the inverse rate from the `CONVERSION_RATE` column and insert the appropriate inverse rate rows into `GL_DAILY_RATES`.

USER_ID: The user ID of the person who is adding rows to the interface table. To determine the user ID for a specific user name, use the following SQL*Plus statement:

```
select user_id
       from fnd_user
      where user.name='<user name>'
```

LAUNCH_RATE_CHANGE: If you want the rate change program to run automatically, enter a 'Y' in the `LAUNCH_RATE_CHANGE` column for *one row* of the rates you are loading. Leave this column blank for the remaining rows. Otherwise, multiple concurrent requests will be launched when only one is required to load all of your rates.

When a daily rate has changed, the rate change program will outdate average translations in those average balance ledgers that use the changed daily rate.

CONTEXT: The descriptive flexfield context.

ATTRIBUTE1 through ATTRIBUTE15: Any descriptive flexfield information associated with the daily rate.

Other Columns

ERROR_CODE: The text of the error message you receive if the row in the interface table failed validation. This column is used by the system. No user entry is needed.

USED_FOR_AB_TRANSLATION: This column is used internally by General Ledger when copying rates to `GL_DAILY_RATES`. Do not make any entries in this column.

Related Topics

Entering Daily Rates, page 5-13

Defining Currencies, page 5-9

Defining Conversion Rate Types, page 5-11

Entering Historical Rates

Enter historical rates or amounts for translating actual and budget account balances. You can enter rates for any foreign currency you have enabled.

You can assign historical rates to accounts, either individually or by range. Generally, you enter historical rates only for specific balance sheet accounts. For example, you can use historical rates to translate non-monetary and selected owners' equity account balances.

Note: Usually, if you are performing translation, enter historical rates only for owner's equity accounts. If you are performing remeasurement, enter historical rates for owner's equity accounts as well as for non-monetary balance sheet accounts and income statement accounts related to non-monetary items.

If you have average balance processing enabled for your ledger, you need to enter separate historical rates for standard and average balances for specific balance sheet accounts.

Note: If you change a historical rate after you've already run translation, you must retranslate your account balances for the period whose rate has changed.

You can use the Currency Rates Manager to create, update, review, and download historical rates using a spreadsheet. See: Currency Rates Manager, page 5-57.

Data Access Sets

The Data Access Sets assigned to your responsibility controls whether or not you can create, modify, delete or view the historical rates for your ledger.

Full Read and Write Access: You can create, modify, delete and view historical rates for your ledger if your data access set provides full read and write access. The following lists the three types of full access:

1. Ledger data access set provides read and write access to the full ledger.
2. Balancing Segment Value data access set provides read and write access to all balancing segment values for a ledger using the All Values checkbox.
3. Management Segment Value data access set provides read and write access to all management segment values for a ledger using the All Values checkbox.

Partial Read and Write Access: If you have read and write access to specific balancing segment values and management segment values, you have the following type of access:

1. Partial Read and Write access to specific balancing segment values or management segment values allows you to create, modify, delete and view historical rates for those balancing segment values or management segment values.

Read Only Access: If you have Read Only access to a ledger, balancing segment values or management segment values, you can only view the historical rates for your ledger.

1. Read Only access to the ledger allows you to view all of the historical rates for your ledger.
2. Read Only access to specific balancing segment value or management segment values allows you to view those specific balancing segment values or management segment values.

Prerequisites

- Define and enable your currencies.
- Define your ledger using Accounting Setup Manager.

To enter a historical rate for a specific account:

1. Navigate to the Historical Rates window.
2. Select the ledger you want to enter historical rates for.
3. Enter the Target Currency for which you want to enter rates. You can enter any foreign currency as the Target Currency.
4. Enter the Period to which the historical rate applies.
5. Enter the Account to which the rate applies.

You must have read and write access to the ledger, balancing segment value or management segment value to enter a historical rate for the account.

6. Enter either a Rate or Amount.

Note: If a historical amount is assigned to an asset, liability, or owner's equity account (assuming Owner's Equity Translation Rule is set to YTD) then, the historical amount will appear in the rate adjustment column in a Translation Trial Balance report.

If a historical amount is assigned to a revenue, expense, or owner's equity account (assuming Owner's Equity Translation Rule is set to PTD) then, the historical amount will appear in the corresponding debit or credit activity columns of a translation trial balance report.

See: Translating Balances, page 5-36 for a discussion of how General Ledger determines the translated balance from the rate or amount you enter on the Historical Rates window.

Note: Data entry for historical amounts in this window assumes you are entering a credit amount (a positive number for a credit amount, a negative number for a negative credit amount).

7. (Optional) If you have average balance processing enabled, choose a Usage type to apply the rate to Standard or Average balances.

Note: You can use the Assign by Ranges window to define the same rate for both standard and average balances.

Note: If average balance processing is not enabled in your ledger, the usage field will not appear in the Historical Rates window.

8. Select Historical as the Rate Type. General Ledger overrides the period-end rate, if one exists, with rates associated with this type.

Note: If you have average balance processing enabled, General Ledger will automatically enter Historical as the Rate Type.

9. Save your work.
10. Produce a Historical Rates Listing to see your historical rates, amounts and weighted-average rates.

To enter a historical rate for a range of accounts:

1. Navigate to the Historical Rates window.
2. Select the ledger or reporting currency you want to enter historical rates for a range of accounts
3. Enter the Target Currency for which you want to enter rates. You can enter any foreign currency as the Target Currency.
4. Choose the Assign by Ranges button.
5. Enter the Period, Rate or Amount, and Rate Type just as you would for individual

accounts. You must have read and write access to the ledger, balancing segment value or management segment value to the accounts to enter historical rates for the range of accounts.

Note: If you have average balance processing enabled, the Rate Type field will not appear.

Note: Data entry for this window assumes you are entering a credit amount (a positive number for a credit amount, a negative number for a negative credit amount).

6. (Optional) If average balance processing is enabled, select a Usage type to apply the rate to Standard, Average, or Standard & Average balances.
7. Enter an account Low and High for the range you want to assign the defined rate. You can assign the same rate to multiple account ranges.
8. Choose OK when you have entered all the ranges for the period, rate, and rate type.
9. Save your work. General Ledger runs a concurrent process to assign historical rates to the accounts in the designated ranges.
10. Produce a Historical Rates Listing to see your historical rates, amounts and weighted-average rates.

Automatically Assigned Rate Types

If you translate an owners' equity account for which you have not entered a historical rate for the period and to-currency, or an asset or liability account for which you have entered a previous historical rate, General Ledger automatically creates a historical rate and assigns it one of the rate types listed below. The information below also describes how General Ledger derives the historical rate it uses for the period and to-currency:

Prior: General Ledger uses the most recently entered historical rate or amount for your balance sheet accounts, and assigns it the rate type Prior.

Note: General Ledger does not automatically roll forward historical rates for the income statement accounts.

Period: If you have never defined a historical rate or amount for an owners' equity account, General Ledger uses:

- The assigned period-average rate type if the profile option GL: Owners Equity Translation Rule is set to PTD.

- The assigned period-end rate type if the profile option GL: Owners Equity Translation Rule is set to YTD.

Calculated: This rate type is only used when the profile option GL: Owners Equity Translation Rule is set to YTD. It is only applicable to the retained earnings account in the first period of each fiscal year. General Ledger calculates a historical rate for the retained earnings account and assigns it the rate type Calculated. This happens regardless of whether a historical rate has been previously defined for the retained earnings account.

Related Topics

Defining Ledgers, *Oracle General Ledger Implementation Guide*

Using Period-End and Period-Average Rates in Translation, *Oracle General Ledger User's Guide*

Historical Rates Listing, *Oracle General Ledger User's Guide*

Overview of Multi-Currency Accounting, page 5-3

Overview of Average Balance Processing, *Oracle General Ledger User's Guide*

Revaluing Balances

Use the Revaluation window to define, run, update, and delete revaluations for foreign currency-denominated balances. Revaluation launches a process that revalues the ledger currency equivalent balances for the accounts and currencies you select, using the appropriate current market rate for each currency. Resulting gain or loss amounts are posted to the gain/loss or cumulative translation adjustment accounts you specify and balanced by balancing segment values. This process creates a revaluation batch containing separate journal entries for each revalued foreign currency.

If you use Reporting Currencies (journal or subledger level), see: Revaluation, *Multiple Reporting Currencies in Oracle Applications*.

You can revalue a single account or ranges of accounts, for both income statement and balance sheet accounts. Income statement accounts are revalued on the basis of their period-to-date or year-to-date balances, in accordance with the Income Statement Accounts Revaluation Rule profile option (See: Setting General Ledger Profile Options., *Oracle General Ledger Reference Guide*). Balance sheet accounts are always revalued on the basis of their year-to-date balances.

When you revalue balances in an average balance ledger, General Ledger only revalues standard balances. When you post the revaluation journal entries to update your standard balances, the system recomputes your average balances automatically.

If you use Reporting Currencies, revaluation journal entries generated and posted in your primary ledger are automatically generated, converted, and posted to each of your reporting currencies.

Secondary Tracking Segment

You can use the secondary tracking segment to track revaluation results using the primary balancing segment and secondary tracking segment. Revaluation gain or loss amounts will be posted to the gain/loss or cumulative translation adjustment account you specify and balanced by each balancing segment value and secondary tracking segment value pair. See: Secondary Segment Tracking, *Oracle General Ledger User's Guide*.

If you use Reporting Currencies, see: *Reporting Currencies User Guide*.

Note: Secondary tracking segment support is not available for average daily balance enabled ledgers. To track revaluation using the cost center segment as the secondary tracking segment in an average balance enabled ledger, set the profile option GL Revaluation: Tracking by Cost Center to Yes.

Note: For non-average daily balance ledgers that require cost center tracking for revaluation but not overall secondary segment tracking, set this profile option to Yes. Otherwise, set this profile option to Null.

Defining, Saving, and Running Revaluations

You can define new revaluations, update existing revaluations and delete revaluations. You can launch any revaluation from the Revaluation window or you can launch saved revaluations and revaluations grouped into Request Sets from the Submit Request window.

- To define a new revaluation, complete the fields in the revaluation window and save your work.
- To update an existing revaluation, query a revaluation, change the fields in the window as necessary and save your work.
- To delete a revaluation, query a revaluation and choose View>Delete from the menu.
- To run a revaluation from the Revaluation window, enter a new revaluation or query a revaluation and choose the Revalue button. Complete the Ledger, Period, Effective Date, and Rate Date fields in the Submit Request window.

Note: Ledgers sharing a common chart of accounts can share revaluations.

- To run a revaluation or group of revaluations from the Submit Request Window, choose the Revaluation program. In the Parameters window that appears, specify the Revaluation Name or Request Set Name for a group of revaluations, Period, Effective Date, and Rate Date.

Grouping Revaluations into Request Sets/Scheduling

You can group revaluations into Request Sets. For example, you can group revaluations into a set to run sequentially or in parallel. You can also schedule revaluations and revaluation sets to run periodically or daily.

- To schedule a revaluation or Request Set, run your revaluation or Request set. In the Parameters window choose the Schedule button. If you select Periodically or Specific Days in the Schedule window, you must enable the Increment checkbox. This will increment the Effective Date and Rate Date. The period will increment only at the end of the period. If you do not enable the Increment checkbox, the Effective Date and Rate Date will not change.

See: *Oracle Applications User Guide* for more information on creating Request Sets and for scheduling options.

PTD Revaluation for Income Statement Accounts

You can use the profile option, GL Income Statement Accounts Revaluation Rule, to specify whether you want to revalue income statement accounts using period-to-date (PTD) or year-to-date (Y-T-D) balances. If you choose to revalue PTD balances for income statement accounts, the program continues to appropriately revalue YTD balances for balance sheet accounts. Revaluing the PTD balance of your income statement accounts creates weighted average YTD balances using period rates from each corresponding period against the PTD account balance and produces more accurate results in compliance with SFAS #52 standards.

When you select the PTD option for revaluing your income statement accounts, Revaluation produces two separate journal entries; one that revalues your balance sheet accounts and another for your income statement accounts. You will not need to reverse the PTD revaluation journal entry for your income statement accounts in the subsequent period since that revaluation only applied to last period's activity.

Prerequisites

- Define accounts for unrealized gains and unrealized losses. You can use the same account for both.
- Define a revaluation rate for each currency for each period or date for which you want to run revaluation.
- If you use Reporting Currencies, define a Cumulative Translation Adjustment account for your ledger.

To revalue your account balances:

1. Navigate to the Revaluation window.
2. Enter the Revaluation Name.
If you do not specify a Revaluation Name, revaluation creates the following name: <Revaluation> <Date> <Time>. For example, Revaluation 14-Oct-2002 10:54:26.
3. (Optional) Enter a Description for your revaluation.
4. (Optional) Enable AutoPost Revaluation. If enabled, the revaluation journal is automatically posted when the revaluation process completes.
5. (Optional) Select the Enable Security checkbox to apply Definition Access Set security to your revaluation definition.

Definition Access Sets are an optional security feature that allows you to control access to your General Ledger definitions. For example, you can prevent certain users from viewing, making changes, or using your revaluation definition.

If you do not enable security, all users will be able to use, view, and modify your revaluation definition.

If the Assign Access function is available for your responsibility, the Assign Access button will be enabled once you check the Enable Security check box. Choose the Assign Access button to assign the definition to one or more Definition Access Sets with the desired privileges. For more information, see Definition Access Sets, *Oracle General Ledger User's Guide*.

If the Assign Access function has been excluded from your responsibility, you will not be able to view the Assign Access button in Revaluation window. You can still secure the revaluation definition by checking the Enable Security check box, but only Definition Access Sets that are AutoAssigned will be automatically assigned to this revaluation definition. See your System Administrator for more information on Function Security.

6. Choose a Currency Option.
 - *All Currencies*: all balances for which the appropriate conversion rates are defined will be revalued.
 - *Single Currency*: only balances for the specified currency will be translated. You must choose a currency from the List of Values in the Currency field.
7. Choose Rate Options.
 - *Daily Rates*: Revaluation will use daily rates of the specified Type to revalue balances. You must choose a type from the Type field.

Type: Revaluation will use this type when you select Daily Rates. The List of Values includes all defined conversion rate types, except User, that are not secured with definition access sets or that are assigned to your definition access sets with Use privilege.

- *One-Time:* Revaluation uses the specified rate to revalue balances. You must specify a rate in the Rate field. One-Time is only available if you choose Single Currency in the Currency Options region.

Rate: Revaluation will use this rate when you select One-Time. Enter a rate greater than 0.

Note: If different rates are required for different ranges of accounts, for example spot rates for balance sheet accounts and average rates for income statement accounts, define separate revaluations for each class of accounts, using a different rate type for each.

8. Enter accounts for the Unrealized Gain and Unrealized Loss accounts. The account can be the same for both fields. The company segment appears blank and is not required.

All debit revaluation adjustments are offset against the unrealized gain account and all credit adjustments are offset against the unrealized loss account. If the same account is specified in both fields, the net of the revaluation adjustments is derived.

9. Complete the fields in the Revaluation Ranges region. For an example, see: Revaluation Example, page 5-33.

- *Account Low:* specify the low end of the range of accounts you want to revalue.
- *Account High:* specify the high end of the range of accounts you want to revalue.

Note: Expand Parent Company and Expand Natural Account fields are automatically checked when you specify the same parent account in the low and high account ranges. These are display only fields. See: Revaluation Example., page 5-33

10. Choose Revalue. Your revaluation is automatically saved and the Submit Request window appears. Complete the following fields:

- **Ledger/Ledger Set:** choose a ledger or ledger set for this revaluation.

If a ledger is selected, the ledger currency cannot be the same as the revalued currency. Revaluation is not generated if the ledger currency is the same as the revalued currency. If a ledger set is selected, revaluation submits a concurrent

request for each of the ledgers in the ledger set. If the revalued currency is the same as the ledger currency, revaluation is not generated for that ledger.

If you use reporting currencies (journal or subledger level), you can choose a reporting currency for this field.

- **Revaluation:** the name defaults to the revaluation definition from the Revaluation form.
- **Period:** choose a period from the List of Values. Only open periods appear in the List of Values.

- **Effective Date:** the default effective date that appears is based on the Period specified. You can change this to any date. The default effective date used is based on the following rules as compared against the system date.

Previous Period: the date will default to the last day of that period.

Current Period: the date will default to the current system date.

Future Period: the date will default to the first day of the period.

If you enter an effective date outside the Period, the effective date will be adjusted to match the specified period when you run Revaluation.

- **Rate Date:** the default date that appears is the same as the effective date.

Changes to the Rate Date field have no effect if you previously specified a rate type of Period or One-Time. If you are using daily rates, you can specify any date.

Note: The profile option GL Revaluation: Days to Roll Forward Daily Rates will roll the daily rate forward for revaluation if you do not define a Daily Rate. See: GL Profile Options., *Oracle General Ledger Reference Guide* Revaluation will only use rates defined for the specified rate date. If a rate is used in this manner, revaluation completes with a warning and the rate is detailed on the Revaluation Execution report.

11. Choose Submit. General Ledger launches a concurrent process to revalue your account balances. The process names your revaluation batch in the following format: Revalues <Period Name> <Concurrent Request Date> <Unique Identifier Number>; for example, Revalues SEP-02 30-SEP-2002 8884.
12. Use the Revaluation Execution Report to review the status of your revaluation. General Ledger automatically generates this report when you run Revaluation.
13. Post the revaluation journal batch if you do not have AutoPost enabled.

Note: Data Access Set security is enforced when revaluation is executed. You must have read and write access to the ledger, or read and write access to the revalued balancing segment values or management segment values. Revaluation only revalues account ranges that it has read and write access to. Accounts combinations with read only access or no access to the balancing segment values or management segment values are ignored.

You must have read and write ledger, balancing segment values or management segment values access to your unrealized gain and loss accounts, otherwise revaluation results in an error.

Note: If you use Reporting Currencies, you must revalue your ledger and each of your reporting currencies (journal or subledger transaction level). Revaluation journal entries generated and posted in your ledger are automatically generated, converted, and posted to each of your reporting currencies.

See: Revaluation, *Oracle General Ledger User's Guide*.

Note: General Ledger conforms to multiple national accounting standards, including SFAS #52 and IAS 21--regarding translation, remeasurement, revaluation, and reporting of foreign currency-denominated balances.

Summary of Revaluation Program

The following summarizes revaluation results, depending on the currency option you choose from the Revaluation window

Single Currency - Revalues standard balances denominated in the selected currency for the selected period and range of accounts. Currency is revalued using the rate defined in the Revaluation window.

All Currencies - Revalues all standard balances denominated in a currency other than your ledger currency, for the selected period and range of accounts. Currencies are revalued using the rate defined in the Revaluation window.

Revaluation Example

The tables below illustrate Revaluation results based on:

- How you enter the range of accounts to be processed
- The parent/child relationships in the range of accounts specified

The examples also illustrate when the Expand Parent Company and Expand Natural Account fields are enabled by the system (when the checks appear in the checkboxes).

Your data access set must provide read and write access to the ledger, or read and write access to the revalued balancing segment values or management segment values.

Assume the following accounts contain foreign currency journal entries. The first segment represents the company balancing segment. The third segment represents the natural account segment.

01.010.1110.0000.100

02.010.1110.0000.100

04.020.1520.0000.000

06.020.3310.0000.000

01.020.2370.0000.000

03.010.2370.0000.000

04.020.2450.0000.000

07.020.3100.0000.000

Parent company value 97 contains child values 01, 02, and 03.

Parent company value 98 contains child values 04, 05, 06, 07, 08, and 09.

Natural Accounts 1000, 2000, and 3000 are parent accounts.

The following revaluation ranges are specified in the Revaluation window:

Specified Revaluation Ranges

Account Low	Account High	Expand Parent Balancing Segment	Expand Natural Account
97.000.1000.0000.000	97.999.1000.9999.999	Yes	Yes
98.000.2000.0000.000	98.999.2000.9999.999	Yes	Yes
97.000.3000.0000.000	98.000.3000.9999.999	No	Yes

Note: A check appears in the Expand Parent Balancing Segment and Expand Natural Account checkboxes when the same values are specified for the Account Low and High. A check will not appear when different parent or natural account ranges are specified for the Account Low and High.

The following accounts will be revalued:

01.010.1110.0000.100

02.010.1110.0000.000

04.020.2450.0000.000

No revaluation results will be created for the 3000 account range because different parent company values are used in the Account Low and Account High ranges. When the low and high parent values are not the same, they are treated as child account ranges.

Assume the following revaluation ranges are specified in the Revaluation window:

Specified Revaluation Ranges, natural account

Account Low	Account High	Expand Parent Balancing Segment	Expand Natural Account
01.000.1000.0000.000	99.999.2000.9999.999	No	No
01.000.3000.0000.000	99.999.3000.9999.999	No	Yes

Note: Note: A check appears in the Expand Parent Balancing Segment and Expand Natural Account checkboxes when the same values are specified for the Account Low and High. A check will not appear when different parent or natural account ranges are specified for the Account Low and High.

The following accounts will be revalued:

01.010.1110.0000.100

02.010.1110.0000.000

04.020.1520.0000.000

06.020.3310.0000.000

07.020.3100.0000.000

No revaluation results will be created for the 2000 parent account because different parent account values were specified in the Account Low and Account High ranges. When the low and high parent values are not the same, they are treated as child account ranges. Only child accounts 1000 through 1999 were considered for revaluation.

Related Topics

Entering Daily Rates, page 5-13

Posting Journal Batches, *Oracle General Ledger User's Guide*

Overview of Multi-Currency Accounting, page 5-3

Overview of Average Balance Processing, *Oracle General Ledger User's Guide*

Overview of Reporting Currencies, *Oracle General Ledger User's Guide*

Translating Balances

You can translate your actual and budget account balances from your ledger currency to another currency. You can launch translation from the Translate Balances window or from the Standard Request Submission (SRS) window. Translated balances are stored in balance-level reporting currencies. Balance level reporting currencies are defined in the Accounting Configuration using the Accounting Setup Manager or automatically generated during translation.

If average balance processing is enabled, you can translate both average and standard balances. Run translation after you have completed all journal activity for an accounting period. If you post additional journal entries or change your translation rates after running translation for a period, you must retranslate. Additionally, if you change the account type for an account segment value and want to retranslate your actual account balances, you may need to purge past translations, change the account type assignment, then run translation.

Important: When you first translate a balancing segment value, you establish the initial translation period. You cannot translate a period before the initial translation period for that balancing segment.

If you mark the All checkbox in the Translate Balances window and attempt to translate new balancing segment values, the program will not process any new balancing segments to prevent you from accidentally establishing the wrong initial translation period. If you add one or more new balancing segments, their first translation must be performed independently. After this first translation, you can mark the All checkbox to translate balances for all balancing segment values that have established initial translation periods.

Translation and the Secondary Tracking Segment

When secondary tracking segment support is enabled for Closing and Translation in the Ledger window, translation will calculate translated retained earnings by summing the translated revenue and expense accounts associated with each unique combination of primary segment value and secondary tracking segment value pair. This amount is closed out to the matching detailed retained earnings account. The system also calculates a historical rate for each detail retained earnings account in the case of the YTD equity method of translation.

This behavior assumes you did not define a historical amount for the retained earnings

account. Otherwise, translation will use the user-defined rate or amount.

When a cumulative translation adjustment is required to balance the translation, the cumulative translation adjustment account will be tracked by each unique combination of primary balancing segment value and secondary tracking segment value pair.

Note: We recommend you translate each period sequentially.

Note: Secondary tracking segment support does not apply to an average translation.

Period-to-Date vs. Year-to-Date Translation Rules

General Ledger uses one of two translation rules shown in the table below, depending on the account type being translated:

- For Asset and Liability accounts, General Ledger always uses the Year-to-Date rule.
- For owner's equity accounts, you can choose to use either of these two rules. If you do not choose a rule, General Ledger uses the Period-to-Date rule.
- For income statement accounts, you can choose to use either of these two rules. If you do not choose a rule, General Ledger uses the Period-to-Date rule.

Note: If Historical Rates or Historical Amounts are used, they will override the period-average or period-end rates for all account types.

Translation Rules

Translation Rule	Translation Period Amount
Period-to-Date (PTD) Rule	Translated Period Amount = Period Average Rate X PTD Ledger Currency Balance
Year-to-Date (YTD) Rule	Translated Period Amount = Period-End Rate X YTD Ledger Currency Balance - Beginning Translated Balance

Rates Used for Translation (Equity Method Translation)

For translation or equity method translation, SFAS #52 and IAS 21 require you use the translation rates in accordance with the following table:

Rates Used for (Equity Method) Translation

GL Account Type	Period-End	Period Average	Historic
Monetary Assets, Liabilities	X		
Non-monetary Assets, Liabilities	X		
Revenue, Expense Related to Monetary Items	X (YTD rule)	X (PTD rule)	
Revenue, Expense Related to Non-monetary Items*	X (YTD rule)	X (PTD rule)	
Equity			X

Rates Used for Remeasurement (Temporal Method Translation)

For remeasurement or temporal method translation, SFAS #52 and IAS 21 require you use the translation rates in accordance with the following table:

Rates Used for Remeasurement

GL Account Type	Period-End	Period Average	Historic
Monetary Assets, Liabilities	X		
Non-monetary Assets, Liabilities			X

GL Account Type	Period-End	Period Average	Historic
Revenue, Expense Related to Monetary Items	X (YTD rule)	X (PTD rule)	
Revenue, Expense Related to Non-monetary Items*			X
Equity			X

Important: Period-end and period-average rates types must be assigned when a ledger is initially created and are defaulted when balance level reporting currencies are automatically created for the ledger. However, you can change the period-end and period-average rate types assignment for the balance level reporting currencies before you run translation for the first time.

The daily rate that is defined for the last day of the period is used as the translation rate. If the rate for the last day of the period does not exist, translation searches back within the period until a rate is found. If no rate exists for the period, translation ends in an error.

Historical rates or amounts override period-end and period average rates for all account types. You should not define a historical rate or amount for an account in the Historical Rates window if you want General Ledger to select the period-end or period average rate for the account according to the above tables.

Note: * Income statement items related to non-monetary items include cost of goods sold, depreciation on property, amortization of intangible items, etc.

Translation vs. Remeasurement

The following table summarizes the major differences in General Ledger setup steps for translation and remeasurement.

Note: The two steps below are the only places in the multi-currency setup flow where translation and remeasurement differ. The other steps are the same for the two translation methods and are omitted below.

Translation vs. Remeasurement

Setup Steps	Translation (Equity Method)	Remeasurement (Temporal Method)
1. Setup Cumulative Translation Adjustment Account	The account type must be owners' equity. (SFAS #52)	The account type must be revenue or expense. (SFAS #52).
2. Enter Historical Rates in Historical Rates Table	Derive and enter historical rates or amounts for owners' equity accounts only.	Derive and enter historical rates for owners' equity accounts, non-monetary assets and liability accounts, and income statement accounts related to non-monetary items.

Note: The above two steps are the only places in the multi-currency setup flow where translation and remeasurement differ. The other steps are the same for the two translation methods and are omitted.

Cumulative Translation Adjustment Account

When you translate your actual balances into another currency, General Ledger automatically adjusts the balance of the Cumulative Translation Adjustment account by the net difference needed to balance your translation results. If you have multiple companies or balancing entities within a ledger, General Ledger automatically adjusts the balance of the translation adjustment accounts of each company or balancing entity. If secondary tracking segment is enabled for your ledger, the Cumulative Translation Adjustment will be calculated by each unique pair of balancing segment and secondary tracking segment values. General Ledger does not make adjustments to this account when you translate budget balances.

Data Access Set

Your data access set must provide read and write access to the ledger or to the specific balancing segment value to run translation. If you only have partial read and write balancing segment values access, you can only translate balancing segment values that you have read and write access to. If you have partial read and write or read only access to management segment values, you cannot run translation.

Reporting Currencies

If you are using Journal or Subledger Transaction Level Reporting Currencies, you cannot run translation in these types of reporting currencies.

Prerequisites

- Define a period in your calendar that precedes the first period you want to translate.
- Define a period in your calendar following the period you want to translate.
- Assign the conversion rate types for the period-average and period-end rates to your ledger using Accounting Setup Manager. Enter period-average, period-end and historical rates for your target currency.
- Enter period and historical rates for your target currency.
- Review the setting of the profile option GL: Owners Equity Translation Rule. If necessary, have your system administrator change the setting. See: Notes on Translating Owners' Equity Accounts, page 5-50.
- Review the setting of the profile option GL Translation: Revenue/Expense Translation Rule. The default is PTD. If necessary, have your system administrator change the setting. See: Notes on Translating Revenue/Expense Accounts, page 5-51.
- Review the setting for Secondary Tracking Segment option for your ledger.
- If you are translating budgets, define your source and target budgets.

To translate actual account balances to a foreign currency:

1. Navigate to the Translate Balances window.
2. Select a ledger or ledger set for this translation.
3. (Optional) If average balance processing is enabled in your ledger or ledger set, select a Usage:

Standard: To translate standard balances only.

Average: To translate average balances only.

Both: To translate both standard and average balances.

Usage defaults to Standard.

Note: If average balance processing is enabled, you can translate both average and standard balances. Additionally, translation cannot be run in an Adjusting period for Average Balances.

4. Mark the All checkbox to translate balances for all balancing segment values, or

enter a single Balancing Segment Value for which you want to translate balances.

Your data access set must have full read and write access to the ledger, or read and write access to all of its balancing segment values or management segment values to select All Balancing Segment. If you only have partial read and write balancing segment values access, you can only translate the specific values that you have read and write access to. If you have partial read and write, or read only access to the management segment values, you cannot run translation for that ledger. The following table displays the Balancing Segment options for the different types of data access set privileges.

If you have this Data Access Type	with read and write access of	you can select the following translation balancing segment option
Ledger	All	All Values or Single Value
Balancing Segment Value	All	All Values or Single Value
Balancing Segment Value	Specific	Single Value
Management Segment Value	All	All Values or Single Value
Management Segment Value	Specific	Cannot Run Translation

Note: If you are launching translation from the Standard Request Submission (SRS) window, leave the Balancing Segment parameter empty to select all balancing segment values. The empty balancing segment parameter defaults to All.

Important: Marking the All checkbox translates balances for all balancing segment values that have been previously translated. If you add one or more new balancing segments, their first translation must be performed independently. After this first translation, you can mark the All checkbox to translate balances for all balancing segment values.

5. Select Actual for the Balance Type to translate.
6. Enter the Target Currency to which you want to translate. If you are translating a ledger, you can choose any enabled currency other than your ledger currency. If you are translating a ledger set, you can choose any enabled currency. Ledgers with a currency that is different than the target currency are submitted for translation.

The Target Ledger defaults to the reporting currency with the same target currency if a ledger is selected. The Target Ledger is disabled if a ledger set is selected.

7. Enter the Period of the balances you want to translate.

Important: The Period you enter the first time you translate actual balances will be the earliest period for which you can translate actual balances for any subsequent translations.

8. Choose the Translate button to begin a concurrent process to translate account balances. General Ledger displays the request ID (ReqID).

Note: Translating both standard and average balances generates two separate concurrent requests; one to translate standard balances and one to translate average balances.

You can launch translation for a ledger or ledger set from the Translate Balance window or the Standard Request Submission (SRS) window by running Program – Translate Balances. When you run translation from the Translate Balance window, the program checks if a balance level reporting currency is assigned to the ledger or ledgers in a ledger set, and checks if translation has been previously run for the target currency. General Ledger performs the following actions:

Launch Translation from Translate Balances window.

Note: You cannot submit an average translation using the Standard Request Submission window. Only standard translations are available.

Ledger	Ledger	Ledger
If a reporting currency is assigned	and the first-ever translation period is established	the following action will take place
Yes	Yes	Submits translation
Yes	No	Uses the entered period as the first- ever translated period and submits translation.

Ledger	Ledger	Ledger
If a reporting currency is assigned	and the first-ever translation period is established	the following action will take place
No	No	Creates a balance-level reporting currency that uses the entered period as the first-ever translated period and submits translation.

Ledger Set	Ledger Set	Ledger Set	Ledger Set
If a reporting currency is assigned to each of the ledgers in the ledger set	and the first-ever translation period for the ledger is established	the following action will take place if you select Yes to auto creating reporting currency and setting the initial translation period	the following action takes place if you select No to auto creating reporting currency and setting the initial translation period
Yes	Yes	Not Applicable *will automatically submit translation	Not Applicable *will automatically submit translation
Yes	No	Uses the entered period as the first-ever translated period and submits translation.	Submits translation only for ledgers with assigned balance level reporting currencies that has established the first-ever period for the entered currency.

Ledger Set	Ledger Set	Ledger Set	Ledger Set
If a reporting currency is assigned to each of the ledgers in the ledger set	and the first-ever translation period for the ledger is established	the following action will take place if you select Yes to auto creating reporting currency and setting the initial translation period	the following action takes place if you select No to auto creating reporting currency and setting the initial translation period
No	No	Creates a reporting currency (balance level), uses the entered period as the first-ever translated period and submits translation.	Submits translation only for ledgers with assigned balance level reporting currencies that has established the first-ever period for the entered currency.

If you are launching translation from the Standard Request Submission (SRS) window, the program checks if a reporting currency is assigned to the ledger or ledgers in a ledger set, and checks if translation has been previously run for the target currency. General Ledger performs the following actions:

Launch Translation from Standard Request Submission (SRS) window.

If you select a	with a reporting currency assigned to the ledger or to each of the ledgers in the ledger set	and the first-ever translation period is established	the following action will take place
Ledger	Yes	Yes	Submit translation.
Ledger	Yes	No	Uses the entered period as the first-ever translated period and submits translation.

If you select a	with a reporting currency assigned to the ledger or to each of the ledgers in the ledger set	and the first-ever translation period is established	the following action will take place
Ledger	No	No	Creates a balance level reporting currency and uses the entered period as the first-ever translated period and submit translation.
Ledger Set	Yes	Yes	Submits translation.
Ledger Set	Yes	No	Does not submit translation.
Ledger Set	No	No	Does not submit translation.

Note: Translation defaults the name of the balance level reporting currency to name of the ledger and appends the currency code, for example, Operations (USD). You can update the name of the reporting currency by using the Accounting Setup Manager.

The Currency Translation Options of period-average and period-end rate types assigned to the ledger is used as the default Currency Translation Options for the balance level reporting currency. You cannot update the reporting currency's Currency Translation Options after you have run translation for the ledger. You need to purge the translated balances first before updating the Currency Translation Options for the reporting currency. The Currency Translation Options for the ledger can be updated at any time.

You can only query and report against balance level reporting currencies. You cannot run translation in journal or subledger transaction level reporting currencies.

You can only translate budget for a ledger, not ledger sets.

To translate budget balances to a foreign currency:

1. Navigate to the Translate Balances window.
2. Select a ledger for this translation.
3. Select the All checkbox to translate balances for all balancing segment values, or enter a single Balancing Segment Value for which you want to translate.

Your data access set must have full read and write access to the ledger, or read and write access to all of its balancing segment values or management segment values to select All Balancing Segment. If you have partial read and write access to the balancing segment values, you can only translate balancing segment values that you have read and write access to. If you have partial read and write or read only access to the management segment values you cannot run translation for that ledger. The following table displays the Balancing Segment options for the different types of user responsibility's data access set privileges.

If you have this Data Access Type	with read and write access of	you can select the following translation balancing segment option
Ledger	All	All Values or Single Value
Balancing Segment Value	All	All Values or Single Value
Balancing Segment Value	Specific	Single Value
Management Segment Value	All	All Values or Single Value
Management Segment Value	Specific	Cannot Run Translation

Note: If you are launching translation from the Standard Request Submission (SRS) window, leave the Balancing Segment parameter empty to select all balancing segment values. The empty balancing segment parameter defaults to All.

4. Choose Budget as the Balance Type to translate.
5. Mark the All checkbox to translate balances for all balancing segment values, or enter a single Balancing Segment Value for which you want to translate balances.
6. Enter the Target Currency to which you want to translate. If you are translating a

ledger, you can choose any enabled currency other than your ledger currency. If you are translating a ledger set, you can choose any enabled currency. Only ledgers with a currency that is different than the target currency are submitted for translation.

The Target Ledger defaults to the reporting currency whose currency is the same as the target currency.

7. Enter the Period of the balances you want to translate. You can translate budget balances for any period regardless of the period you choose to translate first.
8. Enter the Source budget whose account balances you want to translate, and the Target budget for which you want to calculate translated account balances. You can translate one source budget into one or more target budgets.

Important: You should not translate more than one source budget into the same target budget for the same period and currency because each source budget translation will override the balances in your target budget.

Note: You can only translate budget amounts that are entered in the ledger currency.

The budget year containing the period you are translating must be open in your source budget.

9. Enter the Period-Average and Period-End rate types you want to use to translate this budget.

Note: If your conversion rate types are assigned to definition access sets, you must have Use privilege to select a conversion rate type.

10. Choose the Translate button to begin a concurrent process to translate account balances. General Ledger displays the request ID (ReqID).

Note: Secondary Tracking with the Closing and Translation option enabled does not apply to translation of budget balances.

Related Topics

Defining Calendars, *Oracle General Ledger Implementation Guide*

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Using Period-End and Period-Average Rates in Translation, *Oracle General Ledger User's Guide*

Defining Budgets, *Oracle General Ledger User's Guide*

Ledger Options, *Oracle General Ledger Implementation Guide*

Overview of Multi-Currency Accounting, page 5-3

Notes on Translating Average Balances, page 5-53

Overview of Reporting Currencies, *Oracle General Ledger User's Guide*

Overview of Average Balance Processing, *Oracle General Ledger User's Guide*

Notes on Translation with Historical Rates and Amounts

If you have defined historical rates or amounts in the Historical Rates window, General Ledger will select one of two amounts that is used to arrive at a translated balance for your account:

Account Balance: General Ledger uses the historical amount you've provided or translates the account using the historical rate you've provided, and uses the resulting amount as the YTD translated account balance.

Net Activity: General Ledger uses the historical amount you've provided or translates the account's net period activity using the historical rate you've provided, and uses the resulting amount as the translated net period activity for the account. The amount is added to the previous period's translated balance to arrive at the current period's translated balance.

The calculation used depends on whether the account to which the historical rate or amount applies is a revenue/expense, asset/liability, or owners' equity account as well as the translation rule selected.

Asset/Liability: The historical amount becomes the YTD translated balance for the account. If historical rate is used, it is applied against the ledger currency YTD balance.

Owners' Equity: If the profile option GL: Owners Equity Translation Rule is set to PTD, the historical amount is treated as translated net activity for the period. If historical rate is used, it is applied against the ledger currency period net activity. If the profile option is set to YTD, the historical amount becomes the YTD translated balance for the owners' equity account. If historical rate is used, it is applied against the ledger currency YTD balance

Revenue/Expense: If the profile option GL Translation: Revenue/Expense Translation Rule is set to PTD, the historical amount is treated as translated net activity for the period. If historical rate is used, it is applied against the ledger currency period net activity. If the profile option is set to YTD, the historical amount becomes the YTD translated balance for revenue and expense accounts. If historical rate is used, it is applied against the ledger currency YTD balance.

Related Topics

Entering Historical Rates, page 5-23

Notes on Translating Owners' Equity Accounts

General Ledger translates owners' equity accounts in accordance with SFAS #52 and IAS 21, using historical rates or amounts.

Tip: Historical rates tend to be more precise than period-end rates with respect to owners' equity accounts. Therefore, if you translate your owners' equity accounts without defining a historical rate, General Ledger creates a message in the log file stating it used a calculated or period-end rate to perform translation. If this message is created in your log file, we suggest that you define a historical rate and retranslate your balances using that rate.

See: Automatically Assigned Rate Types, page 5-26.

Translating Retained Earnings Account

The retained earnings account at the beginning of a fiscal year is not translated like other accounts in GL. It is translated using the following formula:

Beginning translated retained earnings balance in new fiscal year =

(Sum of all translated revenue balance at the end of prior year -)

Sum of all translated expense balance at the end of prior year +

Translated ending retained earnings balance at the end of prior year.

If you translate owners' equity accounts using the YTD rule, then in the first period of each new fiscal year, General Ledger populates a historical rate with rate type Calculated for the retained earnings account in the historical rates table. It is the ratio of the beginning translated retained earnings account balance to the beginning ledger currency retained earnings account balance. In the case of the YTD rule, user-defined historical rates or amounts are not rolled forward across fiscal years for the retained earnings account. If you translate owners' equity accounts using the PTD rule, General Ledger does not calculate the historical rate for the retained earnings account.

Restating Previously Translated Balances

If you change the translation rule for your owner's equity account, you should restate your previously translated balances. Equity accounts will be translated using the new rule for new translations only. Previously translated equity account balances will not change.

To choose the translation rule to use for owners' equity accounts:

1. Review the setting for the profile option GL: Owners Equity Translation Rule. There are two possible settings:

PTD: Owners' equity is translated using the Period-to-Date rule.

YTD: Owners' equity is translated using the Year-to-Date rule.

2. Have your system administrator set the profile option to the method your organization uses for translating owners' equity.

Note: If you do not maintain historical rates in your ledger, General Ledger will create them for each period for which you translate your owners' equity accounts, using: – The assigned Period-average rates if you use the PTD rule. – The assigned Period-end rates if you use the YTD rule.

To restate your previously translated owner's equity balances after switching the translation rule:

1. Purge the old translated balances for each period to be restated.
2. Change the GL: Owners Equity Translation Rule profile option to the desired setting.
3. For each period to be restated, use the Historical Rates window to delete the rates used to translate owners' equity accounts, as follows:
 - **Retained Earnings:** Delete any non-Historical Type Rates.
 - **Other Owners' Equity accounts:** Delete all assigned Period-Average or Period-End rates.
4. Run translation. Your owners' equity balances will now be translated using the new rule.

Note: Review the historical rates and amounts you have defined to determine if these are still applicable with the change in equity translation rule.

Notes on Translating Revenue/Expense Accounts

The profile option GL: Translation: Revenue/Expense Translation Rule lets you use two translation methods when translating revenue and expense accounts.

When the profile option is set to PTD, the PTD translation rule is applied.

When the profile option is set to YTD, the YTD translation rule is applied.

If you have a business requirement to use both translation methods throughout the year, such as the PTD method during the year for managerial reporting and the YTD method at year-end for legal reporting, you should define additional currencies used solely for translation purposes.

For example, if you must translate balances to Japanese Yen using both PTD and YTD translation methods, define an additional JPY currency called JPYTRANS. The additional currency represents the Japanese Yen Translation currency used as an alternative currency representation of translation.

When you run translation each period, set the profile option to PTD, and run translation for the JPY currency only to use the PTD rule. Then, change the profile option to YTD and re-run translation for the same period using the JPYTRANS currency. This allows you to maintain both types of balances of both currencies that use different translation methods.

Note: You must remember to change the profile option *before* running translation.

To choose the translation rule used for revenue and expense accounts:

1. Review the setting for the profile option GL Translation: Revenue/Expense Translation Rule. There are two settings:

PTD: Revenue and expense accounts are translated using the Period-to-Date rule and assigned period-average rates.

YTD: Revenue and expense accounts are translated using the Year-to-Date rule and assigned period-end rates.

2. Have your system administrator set the profile option to the method your organization uses for translating revenue and expense accounts.

Restating Previously Translated Balances

If you change translation rules for your revenue and expense accounts, you should restate your previously translated balances. Revenue and expense accounts will be translated using the new rule for new translations only. Previously translated revenue and expense balances will not change.

To restate your previously translated revenue and expense balances after switching the translation rule:

1. Purge the old translated balances for each period to be restated.
2. Change the GL Translation: Revenue/Expense Translation Rule profile option to the desired setting.

3. For each period to be restated, update the assigned period end or period average rates in the Daily Rates window as necessary.
4. For each period to be restated, update any rates defined in the Historical Rates window for revenue and expense accounts as necessary.
5. Run translation for every period starting with the earliest period to have your revenue and expense accounts translated using the new rule.

Related Topics

Setting General Ledger Profile Options, *Oracle General Ledger Reference Guide*

Purging Archived Account Balances and Journals, *Oracle General Ledger User's Guide*

Notes on Translating Average Balances

Following are some notes about how General Ledger translates average balances, the rates used for translation, and changing rate types.

How General Ledger Translates Average Balances

When you choose to translate average balances, General Ledger will translate balances for every day in the period you choose to translate. If you subsequently retranslate, the system will retranslate balances for every day in the period you choose to retranslate.

When you translate average balances, the PATD balance type will be translated automatically, using the appropriate calculated average rates See: Rates Used for Translation, page 5-53. If you have chosen to translate optional amount types, see: Ledger Options. General Ledger also automatically translates the average balance types you have selected (i.e., QATD, YATD, and/or EOD).

The cumulative translation adjustment account is not translated directly. Instead, once all other accounts have been translated at the appropriate rates, a balancing entry is made to the cumulative translation adjustment account.

Note: Secondary Tracking with the Closing and Translation option enabled does not apply to translation of average balances.

Rates Used for Translation

When you translate average balances, General Ledger uses averages of different rates, depending on whether the system is translating a non-historical account or a historical account. A historical account is one for which you have entered a historical rate or amount for the Average usage on the Historical Rates window. Non-historical accounts are those for which you have not entered a historical rate.

Non-historical Accounts

General Ledger will use averages of daily rates for the rate type assigned to the ledger, as shown in the example in the following table:

Day	Daily Rate	Average Rate	PATD Balance	Translated PATD
1	1.250	1.250	2,500.00	3,125.00
2	1.300	1.275	3,000.00	3,825.00
3	1.280	1.277	3,250.00	4,150.25
4	1.290	1.280	3,250.00	4,160.00
5	1.320	1.288	3,300.00	4,250.40

Daily rates for all days, business and non-business, are included when General Ledger computes the average rates used to translate non-historical accounts. If there is no daily rate for a specific date, the system will use the most recently entered daily rate for the appropriate rate type.

Historical Accounts

General Ledger uses a weighted average of the historical rates across the number of periods in the specified range being translated. For example, assume the historical rate is 1.25 for January 1996, 1.40 for February, and 1.45 for March. Quarter average-to-date balances for March 16th would be translated using the following weighted-average rate:

Calculations

Description	Rate	Operand	Days in Month	Rate X Days
January calculation	1.25	X	31	38.75
February calculation	1.40	X	29	40.60
March calculation	1.45	X	16	23.20

Description	Rate	Operand	Days in Month	Rate X Days
column sum total			76	102.55

Average Result

Total Rate X Days	Operand	Sum of Days in Month	Rate to Translate March 16th Average Balances
102.55	/	76	1.349

Note: You can choose to specify historical amounts rather than rates in the Historical Rates window. General Ledger will calculate, in the same manner that historical rates are calculated, a weighted historical amount to use for translation.

If you define a historical rate or amount in one period, but not in a subsequent period, General Ledger will automatically roll forward the historical rate or amount from the previous period. This is true for all accounts; not just equity accounts.

If you have never defined a historical rate or amount for an account, General Ledger treats the account as non-historical and translates the average balances using an average of daily rates. This is also true for equity accounts, however, General Ledger will warn you in this instance.

Changing Rate Types

Under certain circumstances, you can change the rate type used to translate an account's average balances. For example, you might initially treat a particular account as non-historical and translate its average balance using an average of daily rates. In a subsequent period, you may decide that the account should be treated as historical and translated using historical rates or amounts. Or, you may initially translate a historical account using historical rates and later decide to translate using a historical amount.

The rules you need to follow when changing rate types for translating average balances are shown in the table below. If you violate these rules, the translation process will terminate with an error.

RATE TYPE FROM:	RATE TYPE TO:	RULES FOR CHANGING
Average Daily Rate	Historical Rate or Historical Amount	After the first translated period, you can only change in the first period of a year.
Historical Rate or Historical Amount	Average Daily Rate	Delete all historical rates or amounts that have been entered since the first translated period.
Historical Rate	Historical Amount	No special considerations if the change is made in the first period of a year. To change in any period other than the first period, you must delete all historical rates entered since the first translated period, then enter your new historical amounts starting from that first period.
Historical Amount	Historical Rate	No special considerations if the change is made in the first period of a year. To change in any period other than the first period, you must delete all historical amounts entered since the first translated period, then enter your new historical rates starting from that first period.

Related Topics

Translating Balances, page 5-36

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Using Period-End and Period-Average Rates in Translation, *Oracle General Ledger User's Guide*

Overview of Multi-Currency Accounting, page 5-3

Overview of Average Balance Processing, *Oracle General Ledger User's Guide*

Currency Rates Manager

The Currency Rates Manager allows you to manage all your currency rate information in one place. You can:

- Enter Daily Rates.
- Upload Daily Rates or Historical Rates from a spreadsheet to Oracle General Ledger.
- Download Historical Rates to a spreadsheet. You can download for review only or update. If you update, you modify the Historical Rates in the spreadsheet and upload to Oracle General Ledger.
- Review Period Rates and historical rates using a web interface.
- Create Cross Rates. Cross rates are calculated conversion rates based on defined currency rate relationships. General Ledger will calculate cross rates based on a Cross Rate Rule you define.

From the General Ledger Navigator choose Setup > Currencies > Currency Rates Manager. Select one of the following tabs: Daily Rates, Historical Rates, Period Rates, or Rate Types.

It is recommended you use the Microsoft Internet Explorer web browser to utilize the full functionality of the Currency Rates Manager.

Related Topics

Cross Rates, page 5-57

Using the Daily Rates Spreadsheet Interface, page 5-61

Using the Historical Rates Spreadsheet Interface, page 5-62

Overview of Multi-Currency Accounting, page 5-3

Defining Currencies, page 5-9

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Cross Rates

Cross rates are calculated conversion rates based on defined currency rate relationships. General Ledger will calculate cross rates based on a Cross Rate Rule you define.

A Cross Rate Rule is associated with a conversion rate type and consists of a Rate Type, Pivot Currency, and Contra Currencies.

Conversion Rate Type: A parameter that associates contra currencies with the pivot currency.

Pivot Currency: The central currency that interacts with Contra Currencies.

Contra Currencies: Currencies that have a rate relationship with the Pivot Currency.

How Cross Rates are Generated

The following text and tables explain how General Ledger generates cross rates or calculated conversion rates. For the examples below, calculations are rounded to five decimal places.

Establish a Cross Rate Rule:

- Enter a Rate Type.
- Assign a Pivot Currency (USD) to the Rate Type.
- Associate one or more contra currencies with the pivot currency (GBP, CAD, EURO).

Enter daily rates between the pivot currency and the contra currency.

You can also accomplish this using a spreadsheet or SQL*LOADER to populate the GL_DAILY_RATES_INTERFACE table.

Entered Daily Rates

To Contra Currencies	From USD Pivot Currency
To USD Pivot Currency	
To GBP	0.65000
To CAD	1.50000
To EURO	0.90000

When daily rates are entered, the inverse conversion rate is automatically created by the system (see table below).

Inverse Rates

Contra Currencies	From USD Pivot Currency	From GBP	From CAD	From Euro
To USD Pivot Currency		1.53846	0.66667	1.11111
To GBP	0.65000			
To CAD	1.50000			
To EURO	0.90000			

Run the Daily Rates Import and Calculation program.

The Daily Rates Import and Calculation program is automatically run when daily rates updates are applied and saved. When the program is launched, the Currency Rates Manager calculates conversion rates between the contra currencies based on contra currency rate relationships with the pivot currency (see table below).

System Generated Cross Rates

To Contra Currencies	From USD Pivot Currency	From GBP	From CAD	From Euro
To USD Pivot Currency		1.53846	0.66667	1.11111
To GBP	0.65000		0.43333	0.72222
To CAD	1.50000	2.30769		1.66667
To EURO	0.90000	1.38462	0.60000	

Display the results.

You can display all rate relationships for a Conversion Rate Type. Navigate to the Currency Rates Manager's Daily Rates window and perform a query on the Rate Type used in your Cross Rate Rule. The query results list all the rates. By selecting a rate and using the update action, more details about the record will be displayed, including the identification of system generated cross rates.

When Cross Rates are Automatically Generated

Cross Rates are automatically updated when the Import and Calculation program is run:

- When you enter or delete daily rates in the Currency Rates Manager's Create Daily Rates page.
- When you use a spreadsheet to upload daily rates to the GL_DAILY_RATES table.
- When you upload rates or load daily rates from the GL_DAILY_RATES_INTERFACE table to the GL_DAILY_RATES table.

Note: Cross rates are not updated when you enter or delete daily rates in General Ledger windows.

Updating Cross Rate Rules

You can update a Cross Rate Rule at any time, by adding or removing contra currency assignments. When you add a contra currency to a Cross Rate Rule, cross rates are generated only when the Daily Rates Import and Calculation program is subsequently run.

If you remove a contra currency from a Cross Rate Rule, any cross rates generated previously for that contra currency will remain unless you manually delete them.

Once you create a cross rate rule, you cannot revise the pivot currency. Instead, delete the cross rate rule and create a new rule for the rate type and appropriate pivot currency.

When you delete a cross rate rule, any cross rates generated previously for contra currencies associated with the rule will remain unless you manually delete them.

When you delete cross rate rules, or when contra currencies are removed from or added to a cross rate rules after the rule has already been in use, users should review the generated cross rates for the rate type. Changes to the rule are not retroactive and will not affect previously stored cross rates.

Note: The GL Daily Rates: Cross Rates Override profile option governs the behavior of generated cross rates. See: *GL Daily Rates: Cross Rates Override, Oracle General Ledger User's Guide*.

Using Custom Rate Loading Programs

The GL_DAILY_RATES_INTERFACE.AL trigger is disabled upon upgrade to this feature. To continue using the trigger logic, set the GL_CRM_UTILITIES_PKG.ENABLE_TRIGGER:=TRUE.

To enable cross rates logic either call the public Application Program Interface (API)

GL_CRM_UTILITIES_PKG.DAILY_RATES_INPUT or run the Daily Rates Import and Calculation program from the Submit Request window.

See Also: Daily Rates, page 5-13.

Related Topics

Currency Rates Manager, page 5-57

Using the Daily Rates Spreadsheet Interface, page 5-61

Using the Historical Rates Spreadsheet Interface, page 5-62

Overview of Multi-Currency Accounting, page 5-3

Defining Currencies, page 5-9

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Using the Daily Rates Spreadsheet Interface

In the Currency Rates Manager, navigate to the Daily Rates tab and choose the Create in Excel button.

Web ADI is launched. Click through the Integrator page. In the Content page, select none to create a blank spreadsheet or select Text to populate the spreadsheet with values from a text file. The format of the spreadsheet is fixed due to mapping requirements between the spreadsheet and the GL_DAILY_RATES_INTERFACE table.

Details on using the spreadsheet are listed below:

Header Region

Action cell: Enter an Action or use the List of Values. **

Delete: Delete the rows in your spreadsheet from the GL_DAILY_RATES table.

Insert: Insert/Update the rows in your spreadsheet into the GL_DAILY_RATES table.

Spreadsheet Region

All columns in your spreadsheet marked by an asterisk (*) are required fields.

From Currency and To Currency columns: Enter or use the List of Values. **

From Date and To Date columns: Use the date format for your installation.

Rate Type column: Enter or use the List of Values. **

Rate column: Enter a rate.

Inverse Rate: Automatically calculated by the system but updatable.

**To display a List of Values, double-click in the cell or place your cursor in the cell and choose Oracle > List of Values from the toolbar menu.

When the data in your spreadsheet is complete and you are ready to upload, choose Oracle > Upload from the menu. Choose the Parameters button to modify the upload parameters or choose the Upload button.

In the Parameters window, you can choose Automatically Submit Daily Rates Import which automatically runs the Daily Rates Import and Calculation program. This will transfer daily rates from the GL_DAILY_RATES_INTERFACE table to the GL_DAILY_RATES table and automatically generate cross rates.

If you don't choose Automatically Submit Daily Rates Import, run the Daily Rates Import and Calculation program to update the GL_DAILY_RATES table and generate cross rates.

To monitor the status of your upload and daily rates import, choose Oracle > Monitor from the menu.

Related Topics

Currency Rates Manager, page 5-57

Cross Rates, page 5-57

Using the Historical Rates Spreadsheet Interface, page 5-62

Multi-Currency Overview, page 5-3

Defining Currencies, page 5-9

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Using the Historical Rates Spreadsheet Interface

In the Currency Rates Manager, navigate to the Historical Rates tab. Select Create in Excel Spreadsheet from the Actions poplist. Click on the Go button.

Web ADI is launched. Click through the Integrator page. In the Content page, select none to create a blank spreadsheet or select Text to populate the spreadsheet with values from a text file. The format of the spreadsheet is fixed due to mapping requirements between the spreadsheet and the GL_HISTORICAL_RATES table.

Details on using the spreadsheet are listed below:

All fields in your spreadsheet marked by an asterisk (*) are required fields.

Header Region

Target Currency:Enter or use the List of Values. **

Period:Enter or use the List of Values. **

Spreadsheet Region

Account Columns: Enter account information.

Value Type: Enter Amount or Rate or choose from the List of Values.**

Value: Enter the amount or rate.**

Rate Type: Enter Historical. You can only upload historical rates with the rate type Historical.

Usage: Defaults to a value of Standard. You can select Average for an Average Daily Balance Ledger.

**To display a List of Values, double-click in the cell or place your cursor in the cell and choose Oracle > List of Values from the toolbar menu.

Enter account and historical rate information in the spreadsheet. You can also cut and paste accounts and historical rate information from another spreadsheet.

When your spreadsheet is complete and you are ready to upload, choose Oracle > Upload from the menu. Choose the Parameters button to modify the upload parameters or choose the Upload button.

To monitor the status of your upload and daily rates import, choose Oracle > Monitor from the menu.

Downloading Historical Rates

In the Currency Rates Manager, navigate to the Historical Rates tab. Select Review or Update from the Action poplist. Review allows you to view the data only. Update allows you to modify the data. Click on the Go button.

Web ADI is launched. In the Mapping window that appears, make your selections for:

Target Currency: Enter a target currency or choose from the List of Values.

Period: Enter or choose a period from the List of Values.

Rate Type: Enter the Rate Type or use wild cards to return a search list. You can specify Historical, Period, Calculated or Prior rate types.

Balancing Segment Low: Specify

Balancing Segment High: Specify

Exclude Historical Rate or Historical Amount of Zero: Select Yes to exclude historical rates or historical amounts of zero.

Choose next to launch your spreadsheet. The format of the spreadsheet is fixed due to mapping requirements between the spreadsheet and the GL_HISTORICAL_RATES table.

Modify the spreadsheet as you desire.

Enter account and historical rate information in the spreadsheet. You can also cut and

paste accounts and historical rate information from another spreadsheet..

Note: Updated historical rates with rate types of Period, Calculated, or Prior must be modified to a rate type of Historical. You can only upload historical rates with the rate type Historical.

When your spreadsheet is complete and you are ready to upload, choose Oracle > Upload from the menu. Choose the Parameters button to modify the upload parameters or choose the Upload button.

Related Topics

Currency Rates Manager, page 5-57

Cross Rates, page 5-57

Using the Daily Rates Spreadsheet Interface, page 5-61

Overview of Multi-Currency Accounting, page 5-3

Defining Currencies, page 5-9

Entering Daily Rates, page 5-13

Entering Historical Rates, page 5-23

Cash Flow Edits

This chapter discusses the procedure for validating and cleansing your Account table data before you process it to generate transfer pricing results.

This chapter covers the following topics:

- Overview of Cash Flow Edit Rules
- Creating Cash Flow Edit Rules
- Executing Cash Flow Edit Rules

Overview of Cash Flow Edit Rules

Cash Flow Edit rules allow you to verify the accuracy and check the completeness of your Account table data. See: *Performing Cash Flow Edits*, page 2-52.

The procedure for working with and managing the Cash Flow Edits rule is similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Cash Flow Edit rules. See: *Searching for Rules*, page 3-5.
- Viewing and Updating Cash Flow Edit rules. See: *Viewing and Updating Rules*, page 3-7.
- Duplicating Cash Flow Edit rules. See: *Duplicating Rules*, page 3-7.
- Deleting Cash Flow Edit rules. See: *Deleting Rules*, page 3-9.

Ideally, you should create and run Cash flow Edit rules on your Account table data before you submit it to the cash flow engine for processing. See:

- *Creating Cash Flow Edit Rules*, page 6-2.
- *Executing Cash Flow Edit Rules*, page 6-4.

Related Topics

Cash Flow Edit Logic, *Oracle Financial Services Reference Guide*

Standard Navigation Paths, page A-1

Creating Cash Flow Edit Rules

Creating a Cash Flow Edit rule is a one-step process. You define both the attributes that uniquely describe a particular Cash Flow Edit rule and the data to be validated or cleansed by that rule on the Create Cash Flow Edit Rule page.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Conditions	One of the two components that determine the data that will be cleansed by Cash Flow Edit rules. Also known as the Data Filter ID, this field allows you to select a subset of data for processing by selecting a Condition that was previously created. Its default value is blank.
Tables	One of the two components that determine the data that will be cleansed by Cash Flow Edit rules. This field allows you to select the Account tables that need to be included in a Cash Flow Edit process.
Preview Mode	Selecting this check box allows you to view the results of running a Cash Flow Edit Rule before the system updates the underlying records in the Account tables. Its default value is unchecked.
Available Tables	One of the two Shuttle Control windows, it contains the names of the Account tables available for inclusion during a Cash Flow Edit process.

Term	Description
Selected Tables	One of the two Shuttle Control windows, it contains the names of the tables that have already been selected for processing by the Cash Flow Edit process.

1. Navigate to the Cash Flow Edits rule home page.

2. Click Create.

The Create Cash Flow Edits Rule page is displayed.

3. Complete standard steps for this procedure. See: Creating Rules, page 3-6.

Note: At this point, you can input the components to ensure that the data processed by Cash Flow Edits will be clean. If you save the Rule without entering any of the Cash Flow Edit components, the Rule will be saved but no data would be selected for cleansing.

4. (Optional) Select Preview Mode.

Important: If you do not select Preview Mode, data inconsistencies are corrected and a log of the changes is output to the log table.

5. (Optional) Select a Condition.

6. Select the Account tables.

Note: Use the Shuttle Control to select the Account tables that you want to include in the Cash Flow Edit process. You can move Account tables from Available Tables into Selected Tables and vice versa by using Move, Move All, Remove, and Remove All. These tables can also be reordered to change the order of processing.

Initially, the selected tables list is empty. However, during subsequent runs, the selected tables list retains the names of the tables that you selected previously. For example, if you select two tables and save the Cash Flow Edits Rule, the system shows them the next time you open the rule.

A table name shown in the Selected Tables does not appear in the Available Tables.

7. Click Apply.

The Cash Flow Edits rule is saved and the home page is displayed.

Related Topics

Performing Cash Flow Edits, page 2-52

Overview of Cash Flow Edits Rules, page 6-1

Standard Navigation Paths, page A-1

Executing Cash Flow Edit Rules

You execute a Cash Flow Edit rule to check the accuracy and the completeness of your Account table data.

Prerequisites

- Predefined Rules

Procedure:

1. Navigate to the Cash Flow Edits rule home page.
2. Search for a rule, page 3-5.
3. Click Run corresponding to the rule you want to execute. The Cash Flow Edits run page is displayed.

Note: You can view the results of running a Cash Flow Edits rule before the system updates the underlying records in the Account tables, provided you selected Preview Mode while defining it.

4. Click Submit to execute (run) the rule.

The Requests page is displayed.

Important: In case you do not want to run the rule immediately, click Apply to save the settings, and make a note of the Job ID displayed by the application. You can use the Job ID to schedule the execution of the rule on the Process Management tab. See: *About Concurrent Programs and Requests, Oracle Enterprise Performance Foundation User's Guide.*

Procedure to Determine the Object ID of the Cash Flow Edits Process

5. Click Refresh repeatedly on the Requests page until the Phase (status) of the Cash Flow Edits process is displayed as completed.
6. Click on Details. The Requests details page is displayed.
7. Click View Log on the Requests details page.
8. Note the Load Data ID. The Object ID is numerical part of the Load ID. For example, if the Load ID is FTP.RUNBCPROCID11375, then 11375 is the Object ID.

You can use the Object ID to view the results of the Cash Flow Edits process as follows:

- Creating a Condition associated with the Object ID Number. See: *About Conditions, Oracle Enterprise Performance Foundation User's Guide.*
- Querying the FTP_CF_CORRECTIONS table with a Data Inspector rule to view the data. See: *About the Data Inspector and Data Inspector Rules, Oracle Enterprise Performance Foundation User's Guide.*

Important: You can also use the Requests subtab on the Process Management tab to determine the Object ID of the Cash Flow Edits process. See: *About Concurrent Programs and Requests, Oracle Enterprise Performance Foundation User's Guide.*

Related Topics

Performing Cash Flow Edits, page 2-52

Overview of Cash Flow Edits Rules, page 6-1

Standard Navigation Paths, page A-1

User Defined Payment Pattern

This chapter describes the procedure for capturing instrument payment patterns that are too complex to be accommodated in the standard fields of Account tables.

This chapter covers the following topics:

- Overview of User Defined Payment Patterns
- Searching for Payment Patterns
- Creating Payment Patterns

Overview of User Defined Payment Patterns

User defined payment patterns allow you to define custom repayment patterns for products in your portfolio. You can include a payment pattern while generating cash flows for transfer pricing and option cost processing by entering the payment pattern code as the amortization type code for the instrument. See: Defining Payment Patterns, page 2-44.

The procedure for working with and managing Payment Patterns is, to a certain extent, similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Payment Patterns, page 7-2.
- Creating Payment Patterns, page 7-3.
- Viewing and Updating Payment Patterns. See: Viewing and Updating Rules, page 3-7.
- Duplicating Payment Patterns. See: Duplicating Rules, page 3-7.
- Deleting Payment Patterns. See: Deleting Rules, page 3-9.

Related Topics

Standard Navigation Paths, page A-1

Searching for Payment Patterns

Search for a payment pattern to perform any of the following tasks:

- Update
- Duplicate
- Delete

Prerequisites

- Predefined payment patterns

Procedure:

1. Navigate to the Payment Pattern home page. This page is the gateway to all payment patterns and related functionality. You can navigate to other pages relating to payment patterns from this page.
2. Use the Search.
 1. Select the type of search: Code or Description.

Important: You can search for a user defined payment pattern based on either a code number or a description. The code and description parameters are mutually exclusive for search purposes. You must choose one or the other but not both at the same time.

2. Enter the code or description of the Pattern.
3. Click Go.

Only patterns that match the search criteria are displayed.

Related Topics

Defining Payment Patterns, page 2-44

Overview of User Defined Payment Patterns, page 7-1

Creating Payment Patterns

You create payment patterns to capture the repayment behavior of instruments that are too complex to be accommodated through use of the standard account table fields.

Procedure:

1. Navigate to the Payment Pattern home page.

2. Click Create Payment Pattern.

The Create Payment Pattern page is displayed.

3. Enter a code value for the new payment pattern.

Important: The code, also known as an amortization type code, is a numeric internal identifier for the payment pattern. The code value must be a number between 1000 and 29999. The code value you assign to the new pattern must be unique. In addition, the code must be mapped to the appropriate instrument records (AMRT_TYPE_CODE field) to connect the instrument to the appropriate pattern.

4. Enter a brief description for the pattern.

Note: If you do not provide a description for the pattern at the time of its creation, the system automatically generates a generic description: Payment Pattern: Code Value.

5. Select the Payment Pattern Type: Absolute, Relative, or Split.

6. Define the Payment Pattern Term Specifications for payment phases.

The selection of the payment pattern type made in the previous step determines the information you must provide to successfully define that pattern type. See:

- Defining Absolute Payment Patterns, page 7-4.
- Defining Relative Payment Patterns, page 7-7.
- Defining Split Patterns, page 7-10.

Note: The Create Payment Pattern page displays the specifications associated with the Absolute Payment Pattern Type, which is the default Payment Pattern Type value. Should you decide to change this value for any of the other two alternatives, Relative or Split, the system will refresh the payment specifications corresponding to the new Pattern Type. Although you can change your selection of the Pattern Type at any point in this procedure, sometimes this might cause a data loss.

Related Topics

Defining Payment Patterns, page 2-44

Overview of User Defined Payment Patterns, page 7-1

Standard Navigation Paths, page A-1

Defining Absolute Payment Patterns

Absolute payment patterns are commonly used for instruments that are on a seasonal schedule, such as agricultural or construction loans that require special payment handling based on months or seasons.

When working with absolute payment patterns, it is sufficient to define payments for one calendar year. Once the term exceeds a year, the payment schedule will loop until the instrument matures.

Prerequisites

- Select Absolute as the Pattern Type.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Month	This drop-down list allows you to select the month of the payment phase being defined.
Day	Used to specify the day of the month the payment is due.

Term	Description
Delete	Used to delete a row.

1. Select the Payment Type from the drop-down list: Conventional, Level Principal, or Non-Amortizing.

Note: The Payment Type determines the type of information required to successfully define the Payment Phase. See Relation between Payment Phase Attributes and Payment Types, page 7-6.

2. Define the Payment Phases.

Note: A Payment Phase is a set of payment characteristics that defines the time line of the instruments amortization.

1. Select a Month for the pattern.
2. Enter a Date for the pattern.
3. Select the Payment Method.

Note: The available Payment Methods depend on the Payment Type. See: Relation between Payment Method and Payment Types, page 7-7 for details. Payment Methods do not apply to the Non-Amortizing Payment Type.

4. Enter the Value for the Payment Method you selected in the previous step for applicable Payment Types.

Note: If you selected the Interest Only Payment Method in the previous step, the Value field does not apply.

5. Click Add Another Row to add additional Payment Phases to the Pattern and click Delete corresponding to the rows you want to delete.

Important: A Payment Pattern must have at least one valid Payment Phase to be successfully defined. The system raises a warning if you try to save a Payment Pattern with an

incomplete Payment Phase. You can define up to 365 Payment Phases for each Payment Pattern.

3. Click Apply.

The Payment Pattern is saved and the Payment Pattern home page is displayed.

Note: Any empty rows are ignored and not saved with the payment pattern.

Guidelines

When a detail instrument using an Absolute Payment Pattern is processed for as-of-date cash flow processing, the Next Payment Date is internally calculated to determine which Payment Phase should be used. The calculated Next Payment Date is only used for this purpose. The Next Payment Date stored on the instrument record in the Account table is always the date used for processing the initial payment.

The following table describes the relationship between Payment Phase properties and Payment Types.

Relationship between Payment Phase Attributes and Payment Types

	Conventional	Level Principal	Non Amortizing
Month	Yes	Yes	Yes
Day	Yes	Yes	Yes
Payment Method	Yes	Yes	
Value	Yes	Yes	

The following table describes relationship between Payment Method and Payment Types.

Relationship between Payment Method and Payment Types

Payment Method	Conventional	Level Principal	Non-Amortizing
Percentage of Original Balance		Yes	
Percentage of Current Balance		Yes	
Percentage of Original Payment	Yes	Yes	
Percentage of Current Payment	Yes	Yes	
Absolute Payment	Yes	Yes	
Interest Only	Yes	Yes	

Related Topics

Defining Payment Patterns, page 2-44

Creating Payment Patterns, page 7-3

Defining Relative Payment Patterns

You create Relative Payment patterns for instruments that have irregular scheduled payments.

Prerequisites

- Select Relative as the Pattern Type.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Frequency	The frequency of the payment.
Multiplier	The unit of time applied to the frequency. The choices are: <ul style="list-style-type: none">• Days• Months• Years
Repeat	The number of times the Payment Phase should be repeated.
Move Up	Allows you to move a particular Payment Phase row up by one position. Note: The Move Up icon for the first row of the table is always inactive.
Move Down	Allows you to move a particular row down by one position. Note: The Move Down icon for the last row of the table is always inactive.
Delete	Allows you to delete a row.

1. Select the Payment Type from the drop-down list: Conventional, Level Principal, or Non-Amortizing.

The payment type determines the available characteristics for defining the payment amount.

2. Define the Payment Phase.

Note: The payment type determines the type of information required to successfully define the payment phase. See: Relation

between Payment Phase Attributes and Payment Types, page 7-10.

1. Enter the Frequency for each payment phase.
2. Select the appropriate Multiplier for each payment phase.
3. Enter the number of times each Payment Phase should be repeated in the Repeat column.
4. Select the Payment Method.

Note: The available payment methods depend on the payment type. See: Relation between Payment Method and Payment Types, page 7-7 for details. Payment Methods do not apply to the Non-Amortizing Payment Type.

5. Type the Value for the Payment Method you selected in the previous step for applicable Payment Types.
6. Click Add Another Row to add additional Payment Phases to the Pattern and click Delete corresponding to the rows you want to delete.

Important: A Payment Pattern must have at least one valid Payment Phase to be successfully defined. The system raises a warning if you try to save a Payment Pattern with an incomplete Payment Phase. You can define up to 365 Payment Phases for each Payment Pattern.

3. Click **Apply**.

The payment pattern is saved and the Payment Pattern home page is displayed.

Note: Any empty rows are ignored and not saved with the payment pattern.

Guidelines

It is not necessary to set up relative payment patterns for the complete term of an instrument. The payment pattern automatically repeats until maturity date. Suppose a payment pattern is created to make monthly payments for the first year and quarterly payments for the next three years. If you apply this pattern to an instrument record with an original term of five years, the payment pattern wraps around and the fifth year is scheduled for monthly payments.

An easy way to set up payment patterns for instruments with varying original terms is to use the repeater of 999 in the last row of the payment pattern. For example, a payment pattern that pays monthly for the first year and quarterly thereafter, can be set up with two rows. The first row shows 12 payments at one month. The second row shows 999 payments at three months. When this payment pattern is processed it repeats the three-month payment frequency until the maturity date is reached.

The following table describes the relationship between payment phase attributes and payment types.

Relationship between Payment Phase Attributes and Payment Types

Payment Phase Attributes	Payment Types: Conventional	Payment Types: Level Principal	Payment Types: Non-Amortizing
Frequency	Yes	Yes	Yes
Multiplier	Yes	Yes	Yes
Repeat	Yes	Yes	Yes
Payment Method	Yes	Yes	
Value	Yes	Yes	

Related Topics

Defining Payment Patterns, page 2-44

Creating Payment Patterns, page 7-3

Defining Split Payment Patterns

You use a Split payment pattern for financial instruments that make principal payments along two concurrent amortization schedules. Split patterns may be a combination of Absolute and Relative Payment Patterns for example, and contain multiple sets of payment phases under a single amortization code. These patterns could further use a combination of Conventional, Level Principal, and Non-Amortizing Payment Types.

Prerequisites

- Select Split as the pattern type.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Percent	The percent value represents the percentage weight of the time line being defined for the individual payment phases (each row). The sum of the percentage weights must total 100%.

1. Click Create Split.

The Create Term Specifications page is displayed.

2. Select the required Pattern Type.

- Absolute
- Relative

3. Define the Payment Phases or Splits.

Tip: The payment pattern term specifications for different payment phases or splits vary depending on whether you select the Absolute or Relative Pattern Type. You can define the term specifications for the splits following the steps described previously for defining payment phases for these patterns. See:

- Defining Absolute Payment Patterns, page 7-4.
- Defining Relative Payment Patterns, page 7-7.

4. Click Apply to add splits. The Create Payment Pattern page is displayed.

5. Enter the percentage value for each split.

Important: The sum of the percent values of all splits must add up to 100.

6. Click Apply.

The Split payment pattern is saved and the Payment Pattern home page is displayed.

Related Topics

Defining Payment Patterns, page 2-44

Creating Payment Patterns, page 7-3

User Defined Repricing Pattern

This chapter discusses the procedure for working with and managing user defined repricing patterns.

This chapter covers the following topics:

- Overview of Repricing Patterns
- Searching for Repricing Patterns
- Creating Repricing Patterns

Overview of Repricing Patterns

User defined repricing patterns provide a mechanism to capture instrument repricing patterns that are too complex to be accommodated through the use of the standard account table fields. See: *Defining Repricing Patterns*, page 2-48.

The procedure for working with and managing repricing patterns is, to a certain extent, similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Repricing Patterns, page 8-2.
- Creating Repricing Patterns, page 8-3.
- Viewing and Updating Repricing Patterns. See: *Viewing and Updating Rules*, page 3-7.
- Duplicating Repricing Patterns. See: *Duplicating Rules*, page 3-7.
- Deleting Repricing Patterns. See: *Deleting Rules*, page 3-9.

Related Topics

Standard Navigation Paths, page A-1

Searching for Repricing Patterns

Search for a repricing pattern to perform any of the following tasks:

- Update
- Duplicate
- Delete

Prerequisites

- Predefined repricing patterns

Procedure:

1. Navigate to the Repricing Pattern home page. This page is the gateway to all repricing patterns and related functionality. You can navigate to other pages relating to repricing patterns from this point
2. Use the Search.
 1. Select the type of search: Code or Description.

Important: You can search for a user defined repricing pattern based on either a code number or a description. The code and description parameters are mutually exclusive for search purposes. You must choose one or the other but not both at the same time.

2. Enter the code or description of the pattern.
3. Click Go.

Only patterns that match the search criteria are displayed.

Related Topics

Overview of Repricing Patterns, page 8-1

Standard Navigation Paths, page A-1

Creating Repricing Patterns

You create Repricing patterns to capture the repricing behavior of instruments whose rates change according to complex schedules.

Procedure:

1. Navigate to the Repricing Pattern home page.

2. Click Create Repricing Pattern.

The Create Repricing Pattern page is displayed.

3. Type a code value for the new Repricing Pattern.

Important: The code is a numeric internal identifier for the repricing pattern. The code value must be a number between 500 and 998 and the code value you assign to the new pattern must be unique. In addition, the code must be mapped to the appropriate instrument records (ADJUSTABLE_TYPE_CODE field) to connect the instrument to the appropriate pattern.

4. Type a brief description for the pattern.

Note: If you do not provide a description for the pattern at the time of its creation, the system automatically generates a generic description: Repricing Pattern: Code Value.

5. Select the Repricing Pattern Type: Absolute or Relative.

The selection of the repricing pattern type determines the fields that are displayed in the Repricing Events table and the information you must provide to successfully define that pattern type. See:

- Defining Absolute Repricing Patterns, page 8-4.
- Defining Relative Repricing Patterns, page 8-7.

Note: The Create Repricing Pattern page displays the parameters associated with the Absolute repricing pattern type, which is the default repricing pattern type value. If you change this value to Relative, the system refreshes the repricing specifications corresponding to the new pattern type, and any data entered previously is lost. However, a warning message is displayed when

you change the pattern type. The data is discarded only after your confirmation.

Related Topics

Overview of Repricing Patterns, page 8-1

Defining Repricing Patterns, page 2-48

Standard Navigation Paths, page A-1

Defining Absolute Repricing Patterns

The Absolute repricing pattern is used for instruments that are date dependent. Each specific date is a separate event. You need to enter the month and day for each event, except for the initial event.

Prerequisites

- Selecting Absolute as the pattern type.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Month	In conjunction with the Day field, this drop-down menu, allows to you to specify a unique month-day combination for a repricing event.
Day	In conjunction with the Month drop-down menu, this field allows you to specify a unique month-day combination for a repricing event.
Repricing Type	A read-only field, it displays the repricing type, Flat rate or Indexed rate, associated with a particular event.
Update	Allows you to update the definition of specific repricing events.

Term	Description
Delete	Allows you to delete specific rows in the Repricing Events table.

1. Click Create Event.

The Create Event page is displayed.

2. Select the Repricing Type: Flat or Indexed.

The default is Flat. If you select Indexed, the system automatically changes the fields available for entry. See: Indexed Repricing, page 8-6.

Note: You can change your selection of the repricing type at any point in this process. Sometimes it may cause a loss of data. However, a warning message is displayed when you change the repricing type. The data is discarded only after your confirmation.

Flat Repricing

A Flat rate is a specific rate—it is directly input. See: User Defined Repricing Event, page 2-49.

To define a Flat Repricing Event, complete the following steps on the Create Repricing Events page:

1. Enter the Net Rate.
2. Enter the Gross Rate.
3. Enter the Transfer Rate.

Important: You must enter a valid value for at least one of these rate fields.

4. Click Apply.

The Create Repricing Pattern Page is displayed.

5. Specify the required month-day combination for the event.

Note: You cannot specify a month-day combination for the first event as this row is reserved for the initial period.

At this point, you have the option of defining additional events or saving. To

add an additional event, repeat Step 1: Click Create Event, page 8-5. If you want to save the repricing pattern and events, advance to the next step.

Indexed Repricing

An Indexed rate is a set of parameters used to calculate a rate. See: User Defined Repricing Event, page 2-49.

To define an Indexed Repricing Event, complete the following steps on the Create Repricing Events page.

1. Select the Indexed Repricing Type.

2. Click Go.
The fields associated with the Indexed repricing type are displayed.
3. Enter the Interest Rate Code.
Alternatively, select it from the list of values.
4. Enter the Transfer Pricing Interest Rate Code.
Alternatively, select it from the list of values.
5. Enter the Net Margin.
6. Enter the Gross Margin.
7. Enter the Transfer Pricing Margin.
8. Enter the Rate Cap Life.
9. Enter the Rate Floor Life.
10. Enter the Rate Set Lag and select the appropriate Multiplier.
11. Enter the Yield Curve Term and select the appropriate Multiplier.
12. Click Apply.
The Create Repricing Pattern page is displayed.
13. Specify the required month-day combination for the event.

Note: You cannot specify a month-day combination for the first event as this row is reserved for the initial period.

At this point, you have the option of defining additional events or saving. To add an additional event, repeat Step 1 Click Create Event, page 8-5. If you want

to save the repricing pattern and events, advance to the next step.

3. Click Apply.

The repricing pattern is saved and the Repricing Pattern home page is displayed.

Related Topics

Defining Repricing Patterns, page 2-48

Creating Repricing Patterns, page 8-3

Defining Relative Repricing Patterns

The Relative repricing pattern is used for instruments where the repricing is determined by elapsed time since origination. Defining a Relative repricing pattern involves the definition of a series of repricing events applicable to a specific repricing pattern code. You need to specify the length of each repricing period and the number of times that event should occur before calculating the next event in the pattern.

Prerequisites

- Selecting Relative as the pattern type.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Frequency	In conjunction with the Multiplier drop-down menu, this field allows you to specify how often repricing occurs.
Multiplier	The unit of time applied to the frequency. The choices are: <ul style="list-style-type: none">• Days• Months• Years

Term	Description
Repeat	Allows you to specify the number of times a repricing event should be repeated.
Repricing Type	A read-only field, it displays the repricing type, Flat rate or Indexed rate, associated with a particular event.
Update	Allows you to update the definition of specific repricing events.
Move Up	Allows you to move a particular row up by one position. Note: The icon of the first and second rows is not active.
Move Down	Allows you to move a particular row down by one position. Note: The icon of the first and last rows is not active.
Delete	Allows you to delete specific rows in the Repricing Events table.

The steps to create Relative repricing patterns are similar to creating Absolute repricing patterns. See: Defining Absolute Repricing Patterns, page 8-4.

The only difference is that the fields in the Repricing Events table are different. You need to specify the following parameters in the Repricing Events table for a Relative repricing pattern:

- Frequency
- Multiplier
- Repeat

Related Topics

Defining Repricing Patterns, page 2-48

Creating Repricing Patterns, page 8-3

Interest Rate Codes

This chapter discusses the procedure for working with and managing interest rate codes and loading related data such as historical rates and term structure parameters.

This chapter covers the following topics:

- Overview of Interest Rate Codes
- Searching for Interest Rate Codes
- Creating Interest Rate Codes
- Loading Data

Overview of Interest Rate Codes

Oracle Transfer Pricing requires historical interest rates for transfers rate and option cost processing. However, insufficient security types, inconsistent quoting conventions, and lack of liquidity may make gathering comprehensive interest rate information a challenge. Interest Rate Codes (IRCs) allow you to define and manage historical interest rates for transfer pricing purposes. See: *Creating Interest Rate Codes*, page 2-53.

The procedure for working with and managing Interest Rate Codes (IRCs) is, to a certain extent, similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Interest Rate Codes Rules, page 9-2.
- Creating Interest Rate Codes Rules, page 9-3.
- Loading Data, page 9-5.
- Viewing and Updating Interest Rate Codes. See: *Viewing and Updating Rules*, page 3-7.
- Deleting Interest Rate Codes. See: *Deleting Rules*, page 3-9.

Related Topics

Standard Navigation Paths, page A-1

Searching for Interest Rate Codes

Search for an Interest Rate Code to perform any of the following tasks:

- Load Data
- Update
- Delete

Prerequisites

- Predefined rules

Procedure:

1. Navigate to Interest Rate Codes home page.

The Interest Rate Codes home page is the gateway to all Interest Rate Codes and related functionality. You can navigate to other pages relating to Interest Rate Codes from this point.

2. Use the Search.
 1. Enter the Code.
 2. Enter the name of the Code.
 3. Select the Rate Format.
 4. Select the Reference Currency.
 5. Click Go.

Only Codes that match the search criteria are displayed.

Related Topics

Creating Interest Rate Codes, page 2-53

Overview of Interest Rate Codes Rules, page 9-1

Standard Navigation Paths, page A-1

Creating Interest Rate Codes

You create Interest Rate Codes to define and manage historical interest rate data. Unlike other Oracle Transfer Pricing rules, Interest Rate Codes don't have versions.

Procedure:

1. Navigate to the Interest Rate Codes Home page.

2. Click Create Interest Rate Code.

The Create Interest Rate Code page is displayed.

3. Enter the Code.

Note: A numeric value, the Interest Rate Code must be unique across all the interest rate codes in the database.

4. Enter a name for the Code.

5. (Optional) Enter a brief description for the Code.

6. Select the Rate Format for the Code: Zero Coupon Yield or Yield to Maturity.

The default value is Zero Coupon Yield. The selection of the Rate Format will influence the values available for Compound Basis. See: Dependencies between the Rate Format, Compound Basis and Accrual Basis Parameters, page 9-5.

7. Select the Compound Basis from the drop-down list.

The values available in this list depend on the Rate Format. See Dependencies between the Rate Format, Compound Basis, and Accrual Basis Parameters, page 9-5.

8. Select the Accrual Basis from the drop-down list.

9. Select the Reference Currency from the drop-down list.

10. Define the Interest Rate Code Terms. Entering the term points for which you want to store the interest rates in a particular interest rate code involves the following steps:

1. Enter a Term value and select a Multiplier, day, month, or year, for this term point from the drop-down list.

2. (Optional) To add additional term points, click Add Another Row.

Important: There are certain restrictions on Terms. They are:

- The maximum number of Term Points that an Interest Rate Code can have is 731.
- A new daily point will be rejected when its frequency falls within the range of an existing term having a monthly or yearly equivalent. For example, a new term of 28, 29, 30, or 31 days will be rejected when there is an existing term of one month.
- A new monthly term will be rejected when its frequency falls within the range of an existing daily or yearly equivalent. For example, a new term of one month will be rejected when a daily point of 28, 29, 30, or 31 already exists.
- If a new monthly point when divided by 12 overlaps an existing yearly point, it will be rejected. For example, a new term of 36 months duplicates an existing term of three years.
- A new yearly point when divided by 12 must not equal an existing monthly point. For example, a new term of two years cannot equal an existing term of 24 months.

3. (Optional) To delete Terms, select the required row and click Delete.

11. Click Apply.

The Interest Rate Code is saved and the Interest Rate Code home page is displayed.

Important: You can save an IRC once you have defined its attributes and terms. However, you need to load historical interest rate data and term structure parameters before you can use it for transfer pricing and option cost processing. See: Loading Data, page 9-5.

Guidelines

The following table describes the dependencies between the Rate Format, Compound Basis and Accrual Basis parameters.

Dependencies between the Rate Format, Compound Basis, and Accrual Basis Parameters

Rate Format	Accrual Basis	Compound Basis: Simple	Compound Basis: Monthly	Compound Basis: Annual	Compound Basis: Semiannual
Zero Coupon		Yes			
	ACT/360	Yes		Yes	Yes
	ACT/ACT	Yes		Yes	Yes
	30/365	Yes		Yes	Yes
	30/ACT	Yes			
	ACT/365	Yes		Yes	Yes
Yield To Maturity	30/360	Yes			
	ACT/360	Yes			
	ACT/ACT	Yes	Yes	Yes	Yes
	30/365	Yes	Yes	Yes	Yes
	30/ACT	Yes			
	ACT/365	Yes	Yes	Yes	Yes

Related Topics

Creating Interest Rate Codes, page 2-53

Overview of Interest Rate Codes Rules, page 9-1

Standard Navigation Paths, page A-1

Loading Data

Before you can use an Interest Rate Code for transfer pricing and option cost processing, you need to load historical interest rate data and term structure parameters for the code. You can add this data using either the Oracle Transfer Pricing user

interface or Web ADI. See:

- Loading Data Using Oracle Transfer Pricing User Interface, page 9-6.
- Loading Data Using Web ADI, page 9-8.

Related Topics

Creating Interest Rate Codes, page 2-53

Overview of Interest Rate Codes Rules, page 9-1

Standard Navigation Paths, page A-1

Loading Data Using Oracle Transfer Pricing User Interface

Use this procedure to load the following:

- **Historical Rates:** You can enter historical interest rates for various term points as they prevailed on specific dates in the past.
- **Parameters:** You can enter values for Interest Rate Code parameters as they prevailed on specific dates in the past. These parameters are used in modeling and forecasting Interest Rate Codes using the interest rate term structure models widely used in the industry such as Merton Model or Ho and Lee Model.

Prerequisites

- Predefined Interest Rate Codes

Procedure to Load Historical Rates:

1. Navigate to the Interest Rate Codes home page.
2. Click Load Data corresponding to the Interest Rate Code for which you want to load data.
3. Select the required Effective Date Range.
4. Select the Data Type: Historical Rates or Parameters.

Note: Historical Rates is the default data type.

5. Click Go.

Depending on the selected Effective Date Range, Historical Interest Rates are displayed.

Note: The Historical Interest Rates are displayed in a descending chronological order.

6. Click Add Row in the Historical Rates Table.

A new row is displayed at the end of the table. It contains the Term Points that have been previously defined for the latest Effective Date Range.

7. Select the Effective Date using the date selector. Alternatively, you can enter it in the space provided.

8. Define the Rates for Term Point columns.

Note: The valid range of rate values for a Term Point is from -999.999999% to 999.999999% (inclusive).

Note: You do not need to define rate values for every Term Point for every effective date. Oracle Transfer Pricing has the capability to interpolate the blank Term Points at run time.

Note: If no rate information is provided for a particular effective date then that row will be ignored by the application.

9. Click Apply.

The Historical Rates are saved and the Interest Rate Code home page is displayed.

Procedure to Load Parameters:

1. Navigate to the Interest Rate Codes home page.
2. Click Load Data corresponding to the Interest Rate Code for which you want to load data.
3. Select the required Effective Date Range.
4. Select Parameters as the data type.
5. Select the required Effective Date Range.
6. Click Go.

Depending on the Effective Date Range, term structure parameters data entered previously for the relevant date range are displayed.

7. Click Add Row in the Parameter Rates table.
A new row is added. It contains fields for entering the term structure parameters. See: Valid and Default Values of Term Structure Parameters, page 9-8.
8. Select the Effective Date using the date picker. Alternatively, you can type it in the space provided.
9. Enter the Mean Revision Speed.
10. Enter the Long Run Rate.
11. Enter the Merton Volatility.
12. Enter the Vasicek Volatility.
13. Click Apply.
The Term Structure Parameters are saved and the Interest Rate Code home page is displayed.

Guidelines

The following table describes the valid and default values for the parameters.

Valid and Default Values of Term Structure Parameters

Term Structure Parameters	Valid Range	Default Value
Mean Reversion Speed	0 to 10.0	0
Long Run Rate	0 to 999.9999%	0
Merton Volatility	0.01% to 10.0%	0.01%
Vasicek Volatility	0.01% to 10.0%	0.01%

Related Topics

Loading Data, page 9-5

Loading Data Using Web ADI

The Web ADI functionality complements the Oracle Transfer Pricing user interface. It is designed to allow Microsoft Excel-based data entry of historical interest rate and parameter information.

Prerequisites

- Predefined Interest Rate Codes

Procedure:

1. Navigate to the Interest Rate Codes home page.
2. Click Load Data corresponding to the Interest Rate Code for which you want to load data.
3. Select the type of data, Historical Rates or Parameter, you want to load.

Important: The data type determines the columns that will be available on the Web ADI spreadsheet.

4. Select the required effective date range.

Important: You can load Web ADI based rates only from an empty spreadsheet.

5. Click Launch Worksheet to invoke Web ADI.

6. Create new record by entering the effective date and associated data.

Important: In the spreadsheet, the IRC term points are reflected generically such as Term 1, Term 2, and Term 3. Consequently, you should input data in the correct chronological order to ensure that rates are uploaded appropriately. For reference, the IRC term structure is documented in the contextual area at the top of the spreadsheet. Web ADI allows you to edit the column descriptions to reflect the appropriate term point description. For example, you can change 'Term 1' to '1 D' for informational purposes and save the spreadsheet, along with this edit, locally for future use.

7. Select Upload on the Oracle menu of the spreadsheet.

The system performs data validations and the Interest Rate Code home page is displayed.

Important: The Web ADI rate loader does not restrict you from loading rates for an effective date that already exists in the database. The new rates will overwrite the existing rates. The

assumption made is that these rates will be the same for any given effective date. In addition, the Web ADI rate loader allows you to input more than one row with the same effective date for a given IRC in the spreadsheet. In this case, the first occurrence of the effective date is loaded and any subsequent occurrences of the same effective date are ignored.

Related Topics

Loading Data, page 9-5

Rate Index Rules

This chapter describes the steps you need to take to work with and manage the Rate Index Rules.

This chapter covers the following topics:

- Overview of Rate Index Rules
- Defining Rate Indexes

Overview of Rate Index Rules

The Rate Index rule allows you to specify the way you want the application to derive forward interest rates for the Interest Rate Codes used to price the products in your portfolio. This is done by establishing a relationship between a risk-free Interest Rate Code (IRC), called valuation curve, and other interest rate codes or indexes. The Rate Index rule is one of the parameters that you need to define on the Transfer Pricing Process version definition page for option cost processing. See: Transfer Pricing Process Rule and Rate Index Rules, page 2-38.

The procedure for working with and managing the Rate Index rule is similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Rate Index rules. See: Searching for Rules, page 3-5.
- Creating Rate Index rules. See: Creating Rules, page 3-6.
- Viewing and Updating Rate Index rules. See: Viewing and Updating Rules, page 3-7.
- Duplicating Rate Index rules. See: Duplicating Rules, page 3-7.
- Deleting Rate Index rules. See: Deleting Rules, page 3-9.

As part of creating and updating Rate Index rules, you can also define rate indexes. See: Defining Rates Indexes, page 10-2.

Related Topics

Standard Navigation Paths, page A-1

Defining Rate Indexes

Rate indexes are always associated with a version of a Rate Index rule. As a Rule can have multiple versions, you can define various sets of rate indexes under a single Rate Index rule. The Start and End dates of each Version determine the set of rate indexes available at any particular point in time. The definition of rate indexes is part of the create Rate Index rule process in which rate indexes are defined for currency-valuation curve combinations. When you click Finish in the create Rate Index rule process, the Rule and the Version are saved and the Rate Index rule home page is displayed. However, the rate indexes have not yet been defined for any of currency-valuation curve combinations. Typically, you would start defining the rate indexes for currency-valuation curve combinations before clicking Finish.

Prerequisites

- Creating Interest Rate Codes, page 9-3
- Performing basic steps for creating or updating a Rate Index rule, page 3-6.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Valuation Curve	<p>The Valuation curve is used to calculate the future rates of Indexes (IRCs) defined in a Rate Index rule and these future rates are used to calculate option costs. Oracle Transfer Pricing allows you to reference the Valuation curve for a currency during the create process of the Rate Index rule.</p> <p>Although you can select only one valuation curve per currency, you can select the same or different valuation curve for different currencies. Typically, the Valuation curve and the Index Rate curves derived from it have the same Referenced Currency. For example, you will use the US Treasury Yield Curve as the Valuation curve to calculate the forward rates of any US dollar-based Interest Rate Code.</p> <p>However, it is not mandatory that the Reference Currency of the Valuation Curve and the currency of the Index Rate Curves for which forward rates are to be generated should be the same. For example, if a country lacks the liquidity in its credit markets to have a local currency risk free rate curve, you may want to use the risk free rate curve of a liquid market, such as US, as the valuation curve with a different reference currency than that of the Index Rate Curves.</p>

Term	Description
------	-------------

Index Term An Interest Rate Code is made up of one or many term points that denote a particular interest rate yield curve structure. Oracle Transfer Pricing generates future rates for term points in the Interest Rate Code based on an arithmetic formula that has the following components:

- A combination of term point rates from the valuation curve (with a maximum selection of eight terms or elements), which need not be the standard term points as defined in IRC definition of the valuation curve
- A coefficient and an exponent for each of the valuation curve term points
- A single spread per index term point

A formula must be defined for each index tied to an instrument. That formula takes the following form:

$$\text{Index Rate}_t = \sum_{n=1}^8 [\text{Coefficient}_n \times (\text{RFR}_T)^{\text{Exponent}_n}] + \text{Spread}_n$$

Where:

- t is the term point of the index
- T is the term point of the valuation curve
- n is the number of term points of the valuation curve referenced
- RFR is the rate of the specific term point on the valuation curve

To create your formula, you can select up to eight term points (elements) from the risk free curve, each multiplied by a user-defined coefficient and raised to the power of a user-defined exponent. Additionally, you can add a constant spread for each of the term points used in the formula.

-
1. Navigate to the Rate Index Version definition page.
 2. Select the currency you want to work with from the Valuation Curves table.

Important: This table displays a limited amount of rows. If you do not see the currency you are looking for, use the record navigator provided within the table.
 3. Select a Valuation Curve for the currency you selected in the previous step.

Important: Only a single Valuation Curve can be associated with a particular currency. For example, if the Valuation Curve for US Dollars is US Treasury Yield, all US Dollar indexes will be associated with the US Treasury Yield curve.

Ideally, you need to select a risk free rate interest rate structure. Not all the Interest Rate Codes in the application will have the characteristics of a risk free rate curve, but the application will not prevent you from selecting it as a Valuation Curve.

Procedure to Add the Index

4. Click Add Index.

The Define Index page for the selected currency is displayed.

5. Enter the Code or the Description in the text box to narrow the search.

You can use wildcards or leave the fields blank.

6. Click Go.

The application will only display those indexes that match the search criteria and whose reference currency is the same as the currency you selected earlier in Step 3. It will exclude the index, which has been used as the valuation curve, if the valuation curve belongs to the same currency as selection for defining the index, from the search results.

7. Select the Index you want to define.

8. Click Continue.

The Add Index Term Definition page is displayed. The general attributes of the valuation curve, from which the Index Rates are derived, are displayed. This information can be used as a reference when you define the terms.

Procedure to Add Index Term Definitions

Each Index Term Point can be calculated from up to eight elements of the valuation curve. The valuation curve elements specified can be any term point on the yield curve; it is not restricted to the points displayed for the valuation curve.

9. Select the Index Term you want to define.

Not all IRCs have Term Points defined. To successfully define an Index, you must define at least one of its terms. Optionally, you could define one, many or all of the Index Terms. The selection of Index Term is limited to the standard Term Points as defined in the IRC definition.

10. Enter the Spread for the Index Terms.

A Spread is a constant percentage added to the variable rate produced as a result of the Monte Carlo calculations, multiplication with the defined coefficient and raising to the power of the mentioned exponent.

11. Enter the Valuation Curve Term Point and select the multiplier.

12. Enter a coefficient for the element.

If you do not enter a value, the system uses the default value of 1.

13. Enter an exponent for the element.

If you do not enter a value, the system uses the default value of 1.

14. Repeat the last four steps for a maximum of seven more elements.

15. Click Continue.

The Rate Index Version definition page is displayed.

16. Click Apply to save the changes.

The Rate Index rule home page is displayed.

Related Topics

Overview of Rate Index Rules, page 10-1

Standard Navigation Paths, page A-1

Conditional Assumptions

This chapter discusses the procedure for assigning transfer pricing and prepayment methodologies to product-currency combinations using conditional logic.

This chapter covers the following topics:

- Overview of Conditional Assumptions
- Creating Conditional Assumptions

Overview of Conditional Assumptions

Conditional Assumptions allow you to categorize your product portfolio based on common characteristics, such as term to maturity, origination date, and repricing frequency, and assign specific transfer pricing and prepayment methodologies to each category. See:

- Associating Conditional assumptions with Transfer Pricing Rules, page 2-21.
- Associating Node Level and Conditional Assumptions with Prepayment Rules, page 2-35.
- Creating Conditional Assumptions, page 11-1.

Related Topics

Standard Navigation Paths, page A-1

Creating Conditional Assumptions

Conditional Assumptions cannot exist independently. Creating Conditional Assumptions is a subprocess within the general create flow of the Transfer Pricing and Prepayment rules. Once you have created a rule and a version, you can assign transfer pricing and prepayment methodologies to product-currency combinations either

directly or by creating a conditional assumption using conditional logic.

A Conditional Assumption is made up of at least three (IF, THEN, and ELSE) clauses. Each clause is displayed in the user interface as a row. Here is a high-level overview of the steps necessary to create a Conditional Assumption:

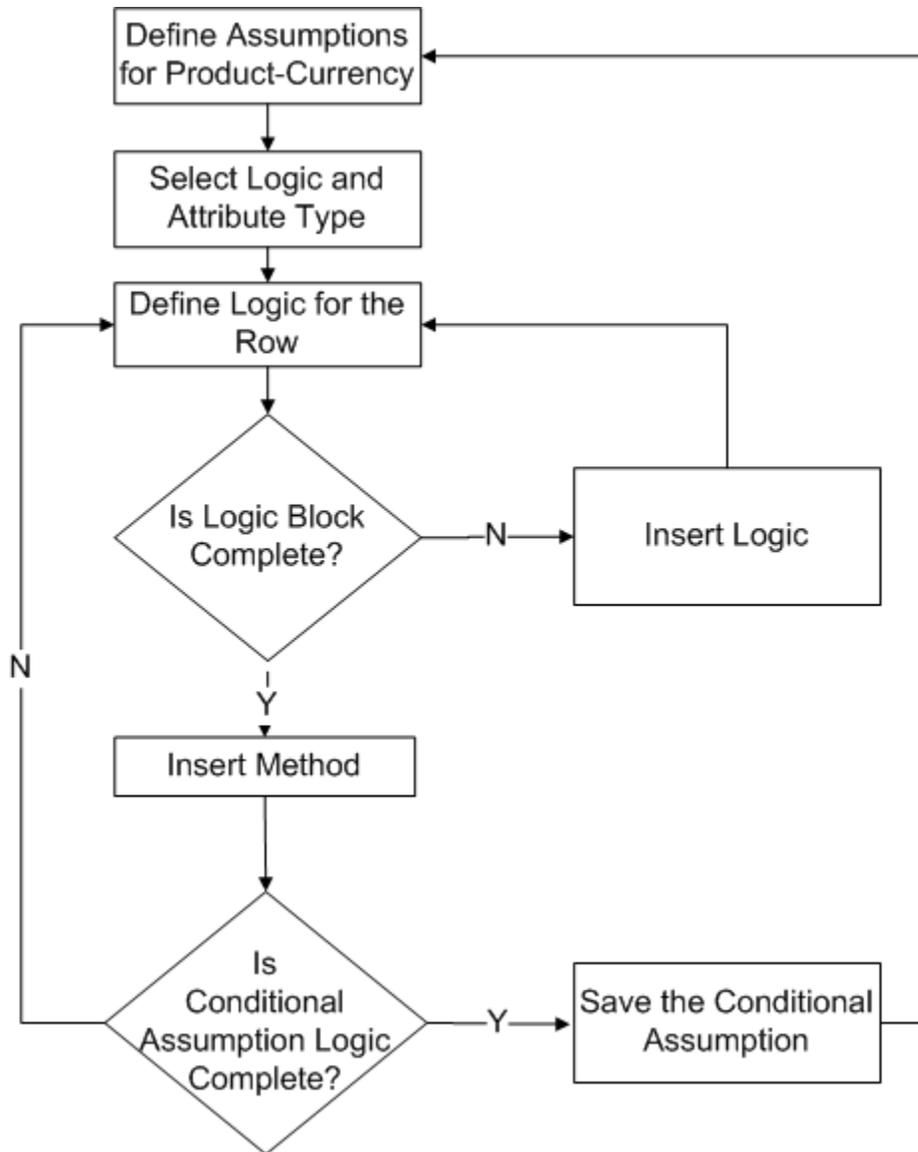
1. Click Create Conditional Assumptions on the Transfer Pricing or Prepayment methodology page.
2. Select the Logic and Attribute types.
3. Define the logic for the IF Block.
4. Complete the block by inserting a transfer pricing or prepayment method (THEN clause).

Note: Alternatively, you can insert an unlimited number of ELSE IF blocks after the IF block. You also have the option to insert an unlimited number of AND/OR clauses within the IF or ELSE IF blocks.

5. To complete the Conditional Assumption, insert an ELSE block. You would associate a transfer pricing or prepayment method to this block to make sure that all the records with the same product-currency combination are transfer priced.
6. Click Apply.

The system validates the logic for the Conditional Assumption. If the logic is correct the Conditional Assumption is saved and the version definition page is displayed. You can then proceed with defining transfer pricing and prepayment methods for other products-currency combinations.

The following graphic illustrates the process for creating conditional assumptions.



Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing or Prepayment rule, page 3-6

Procedure

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Select	This check box allows you to add a new logic clause.
Condition Term	This field lists the type of logic clause for a row (IF, THEN, ELSE IF, AND/OR, or ELSE). This field is read only except when the condition term is AND/OR. When the condition term is AND/OR, this field becomes a drop down list.
Open	This drop-down list allows you to use open parentheses to specify how the engine should interpret the logic for a Conditional Assumption at run time.
Attribute Name	The name of the account table column that is part of the logic. The values available in this column depend on the type of attribute you selected on the Create Conditional Assumption: Select Logic Type page. So make sure that the attribute you enter are of the same type.
Operator	<p>This drop down list provides you with operators to define a relationship between an Attribute Name and a Value.</p> <p>Note: The values available in this list depend on the type of logic and type of attribute you choose on the Create Conditional Assumption: Select Logic Type page.</p>
Value	Input field in which you enter the value for a clause. For Date type attributes, the system displays the calendar control. For Code value type attributes, the system displays a list of values.

Term	Description
Close	This drop-down list allows you to close the piece of logic you initiated previously with an open parenthesis.
Method	A read only field, it displays the transfer pricing or prepayment methods for logic rows with THEN and ELSE Condition Terms.
Update	Allows you to update the logic of a clause.
Delete	Allows you to delete a clause.

1. Navigate to the Transfer Pricing or Prepayment methodology page.

2. Click Create Conditional Assumption.

The Create Conditional Assumption: Select Logic Type page is displayed.

3. Select the Attribute Type.

Your selection determines the type of attribute that will be part of the logic clause. For example, if you choose the Dates attribute type, a date variable will be part of the logic clause that you create.

The following attribute types are available:

- Alphanumeric
- Balances
- Dates
- Dimension
- Numeric
- Rates
- Terms and Frequency

4. Select the Logic Type.

This determines the type of logic that will be part of the logic clause. The following Logic Types are available:

- Specific Value
- Another Column
- Range Values
- List Of Values

The Logic Type you select affects the operators available during the next step in the process, defining the logic on the Conditional Assumptions Logic page. For example, if you select Specific Value, the condition you define on the Conditional Assumptions Logic page will involve the comparison of a specific value, such as the number 10.

The type of logic available to you depends on the Attribute Type you select. See: Logic for Attribute Types: Balances, Rates, Date, Terms & Frequencies and Numeric Attributes, page 11-9 and Logic for Attribute Types: Dimensions and Alphanumeric Attributes, page 11-9.

5. Click Continue.

The Conditional Assumption Logic page is displayed. This is where you define the actual attributes and parameters that make up the logic blocks for the Conditional Assumption.

6. Define Logic for the IF Clause on the Conditional Assumption Logic Page.

1. Select the Attribute Name.
2. Select an Operator.
3. Enter a Value.
4. (Optional) Select Parenthesis:
 - Open.
 - Close (if you selected Open first).

Important: In the absence of any parentheses for grouping, the AND operator does not have any precedence ranking over the OR operator. Consequently, the clauses linked by these operators are processed from left to right. Take for example:

A = 1 OR B = 2 AND C = 3

In the absence of parentheses, these are processed from left

to right as:

$(A = 1 \text{ OR } B = 2) \text{ AND } C = 3$

Instead of AND taking precedence as follows:

$A = 1 \text{ OR } (B = 2 \text{ AND } C = 3)$

At this stage, an initial row of logic, the IF Clause, has been defined. However, you need to take additional steps to complete defining the conditional assumption. See: Procedure to Define Additional Clauses, page 11-7 and Procedure to Validate and Save a Conditional Assumption, page 11-8.

Procedure to Define Additional Clauses

After defining the initial row of logic, the IF Clause, you have three options:

- **Inserting a Method:** You insert a transfer pricing or prepayment method to the block when you have completed defining logic for the Conditional Assumption.

There are two potential outcomes when you decide to insert a method:

- The system adds a new THEN clause if you choose to insert a method after an IF, OR, AND, or ELSE IF clause.
- The system adds an ELSE clause if you choose to insert a method after the THEN clause of the last block of the Conditional Assumption.

See: Inserting a Method, page 11-9

- **Inserting Logic:**

This option allows you to continue building the complexity of the logic within the Conditional Assumption. As with the Insert Method option, there are also two potential outcomes when you decide to insert logic:

- The system appends a new row within the block so that you can choose to define an AND clause or an OR clause. This occurs if you decide to insert logic after an IF, OR, AND, or ELSE IF clause.
- The system adds an ELSE IF clause if you choose to insert logic after a THEN clause.

Inserting logic into a block is optional. Therefore, to create a basic Conditional Assumption with an IF-THEN-ELSE clause, you would not need to insert logic into a block. See: Inserting Logic into a Block, page 11-11.

- **Updating a clause**

Clicking Update Logic Type allows you to change the Field Type for a logic clause on the Create Conditional Assumption: Select Logic Type page. See: Updating a Clause, page 11-12.

Procedure to Validate and Save a Conditional Assumption

When the logic for the Conditional Assumption is complete, click Apply to save the Conditional Assumption. At this point, the system performs a validation on the Conditional Assumption. There are three types of validations that the system performs.

- **Structure:** The structure of the Conditional Assumption is validated. Only the Conditional Assumptions with the correct structure are saved. Among other things, the system will check for the following:
 - The first block of the Conditional Assumption starts with an IF clause and ends with a THEN clause. Between these two blocks you can add as many AND/OR clauses as you want.
 - Any ELSE IF block must be preceded by a THEN clause and it must end with a THEN clause. You can add as many AND/OR clauses between the ELSE IF and the THEN clauses as you want.
 - The final row of the Conditional Assumption is an ELSE clause.
- **Row Logic:** The Row Logic must be valid. For example, any row that describes logic must have a condition term, a field, an operator and a value. Any THEN/ELSE clause must have a method assigned to it.
- **Method Parameters:** You must have successfully defined all the required parameters for a particular transfer pricing or prepayment method for each THEN/ELSE clause in the Conditional Assumption.

If the system cannot validate the Conditional Assumption, it will warn you of the rows that caused the errors. If the validation is successful, you will be returned to the version definition page.

Guidelines

The following table lists the logic available for Balances, Rates, Date, Terms and Frequencies, and Numeric attribute types.

Logic for Attribute Types: Balances, Rates, Date, Terms & Frequencies, and Numeric Attributes

Operator	Input Value	Another Column
=	Yes	Yes
<>	Yes	Yes
>	Yes	Yes
<	Yes	Yes
>=	Yes	Yes
<=	Yes	Yes

The following table lists the logic available for Dimensions and Alphanumeric attribute types.

Logic for Attribute Types: Dimensions and Alphanumeric Attributes

Operator	Input Value	Another Column
=	Yes	NA
<>	Yes	NA

Related Topics

Standard Navigation Paths, page A-1

Overview of Conditional assumptions, page 11-1

Inserting a Method

Use this procedure to insert a transfer pricing or prepayment method to a block of logic when you have completed defining logic for the Conditional Assumption.

Prerequisites

- Performing steps 1 to 6 for creating a Conditional assumption, page 11-3.

Procedure

1. Select the row after which you want to insert the method.

2. Click Insert Method.

The system appends a new row after the row selected. The condition term of the new row depends on the condition term of the row after which you decide to insert the method. For example, inserting a method after an IF clause will cause the system to append a new row underneath it with a THEN condition term. See: Possible Condition Terms for Appended Rows, page 11-10.

You now can assign a specific transfer pricing or prepayment method to the current block of logic.

3. Click Update corresponding to the logic clause.

4. Assign the required methodology.

Note: This procedure is same as the one used for directly assigning a transfer pricing or prepayment methodology. See: Defining Transfer Pricing Methodologies, page 12-3 and Defining Prepayment Methodologies, page 13-3.

5. Click Apply.

The Conditional Assumption Logic page is displayed.

Guidelines

The following table shows the possible condition terms for rows appended as a result of inserting methods.

Possible Condition Terms for Appended Rows: Inserting Method

Preceding Clause	Condition Term for the Appended Row
IF	THEN
AND/OR	THEN
THEN	ELSE
ELSE IF	THEN

Preceding Clause	Condition Term for the Appended Row
ELSE	Not possible to append a row

Related Topics

Creating Conditional Assumptions, page 11-1

Standard Navigation Paths, page A-1

Inserting Logic into a Block

Use this procedure to insert additional logic into a Conditional Assumption.

Prerequisites

- Performing steps 1 to 6 for creating a Conditional assumption, page 11-3.

Procedure

1. Select the row after which you need the new logic to be inserted on the Conditional Assumption Logic page.
2. Click Insert Logic.
The system appends a new row after the row selected. The condition term of the new row depends on the condition term of the row after which you decide to insert the method. See: Possible Condition Terms for Appended Rows: Inserting Logic, page 11-11.
3. Enter the data values for the new logic clause.
See: Step 6, page 11-6 of Creating Conditional Assumptions.

Guidelines

The following table shows the possible condition terms for rows appended as a result of inserting new logic

Possible Condition Terms for Appended Rows: Inserting Logic

Preceding Clause	Possible Terms for inserted Row
IF	AND/OR

Preceding Clause	Possible Terms for inserted Row
AND/OR	AND/OR
THEN	ELSE IF
ELSE IF	AND/OR
ELSE	

Related Topics

Creating Conditional Assumptions, page 11-1

Standard Navigation Paths, page A-1

Updating a Clause

You may want to change the logic of an existing clause. You can do this in any of the following two ways:

- Change the logic of the clause without changing the field type.
- Change both the logic and the attribute type for the clause. For example, you may want to change the Attribute Type from Rates to Balances.

Note: Updating the Attribute Type is optional and is not required to create a basic Conditional Assumption with an IF-THEN-ELSE clause.

Prerequisites

- Performing steps 1 to 6 for creating a Conditional assumption, page 11-3.

Procedure to Change Logic without Changing Attribute Type

- Change the values directly in the Value field on the Conditional Assumption Logic page.
- Click the Attribute Type LOV on the Conditional Assumption Logic page and select another logic attribute. The LOV shows only those logic attributes which are of the same type as the current attribute of the clause.

Procedure to Change both Logic and Attribute Types

1. Click Update Logic Type on the Conditional Assumption Logic page.

The Create Conditional Assumption: Select Logic Type page is displayed.

2. Select the required Attribute Type from the drop-down list.

3. Select the required Logic Type.

4. Click Continue.

The Attribute Type and the Logic Type is updated and the Conditional Assumption Logic page is displayed.

Related Topics

Creating Conditional Assumptions, page 11-1

Standard Navigation Paths, page A-1

Transfer Pricing Rules

This chapter describes the procedure for working with and managing Transfer Pricing rules.

This chapter covers the following topics:

- Overview of Transfer Pricing Rules
- Creating Transfer Pricing Rules
- Defining Transfer Pricing Methodologies

Overview of Transfer Pricing Rules

Transfer Pricing rules allow you to specify methodologies for transfer pricing your product portfolio. A Transfer Pricing rule contains the transfer pricing methodologies defined for all products (Line Item Dimension Members) in a particular product hierarchy. In addition, it contains certain parameters used in defining option cost methodologies. See: *Setting Transfer Pricing Rules*, page 2-3.

The Transfer Pricing rule is a key component of the Transfer Pricing Process rule. The Transfer Pricing Process rule, which is used to launch the transfer pricing process, uses the transfer pricing methodologies contained in the Transfer Pricing rules to generate transfer rates. Consequently, before processing information for a new period, you need to review and validate the assumptions contained in Transfer Pricing rules. If new members are added to the Line Item dimension, you need to update the Transfer Pricing rule by defining appropriate methodologies for the new products.

The procedure for working with and managing the Transfer Pricing rule is mostly similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Transfer Pricing rules. See: *Searching for Rules*, page 3-5.
- *Creating Transfer Pricing Rules*, page 12-2.
- Viewing and Updating Transfer Pricing rules. See: *Viewing and Updating Rules*,

page 3-7.

- Duplicating Transfer Pricing rules. See: Duplicating Rules, page 3-7.
- Deleting Transfer Pricing rules. See: Deleting Rules, page 3-9.

As part of creating and updating Transfer Pricing rules, you can also define transfer pricing methodologies. See:

- Defining Transfer Pricing Methodologies, page 12-3.
- Defining the Redemption Curve Methodology, page 12-9.
- Defining the Unpriced Account Methodology, page 12-10.

Related Topics

Standard Navigation Paths, page A-1

Creating Transfer Pricing Rules

You create a Transfer Pricing rule to define transfer pricing methodologies for your products.

Procedure

1. Navigate to the Transfer Pricing rule home page.
2. Complete standard steps for this procedure. See: Creating Rules, page 3-6.

Important: In addition to the standard steps for creating rules, the procedure for creating Transfer Pricing rule and version involves one extra step. After Standard Step 5, you need to select a product hierarchy. You can define methodologies at any level of the hierarchical product dimension. The hierarchical relationship between the nodes allows inheritance of methodologies from parent nodes to children nodes.

You can also continue defining a methodology at the lowest level of the Line Item dimension without selecting a hierarchy.

Caution: Oracle Transfer Pricing does not support multiple value set hierarchies and hierarchies with multiple tops. In addition, all the hierarchies need to be stored in flattened format for processing by the application. See: Working with Hierarchies, *Oracle Enterprise*

Related Topics

Overview of Transfer Pricing Rules, page 12-1

Standard Navigation Paths, page A-1

Defining Transfer Pricing Methodologies

Transfer pricing methodologies are always associated with a version of a Transfer Pricing rule. As a rule can have multiple versions, you can define various sets of transfer pricing methodologies under a single Transfer Pricing rule. The start and end dates of each version determine the set of methodologies available at any particular point.

The definition of transfer pricing methodologies is part of the Create or Update Transfer Pricing rules process where assumptions about transfer pricing are made for product-currency combinations. When you click Apply in the Create Transfer Pricing rules process, the rule and the version are saved and the Transfer Pricing rule selector page is displayed. However, the transfer pricing methodology has not yet been defined for any of your products at this point. Typically, you would start defining your methodologies for product-currency combinations before clicking Apply.

The Transfer Pricing rule supports definition of assumptions for combinations of two dimensions: Product and Currency. Consequently, before you start defining the transfer pricing methodologies for your products, you need to decide the way you want to input the parameters. You can select any one of the following two paths:

- **All Products for one Currency:** You can define transfer pricing methodologies for your entire product portfolio one currency at a time. Suppose your portfolio comprises of products denominated in two currencies (US Dollar and Japanese Yen) and that you want to specify different transfer pricing assumptions for each product group. Using the *All products for one currency* flow, you can first define assumptions for the products denominated in US Dollars and then proceed with defining assumptions for the Yen-based products.
- **All Currencies for one Product:** You can also decide to define assumptions for all the currencies supported in your system, one product at a time. For example, you can define assumptions for all active currencies for the commercial loans before proceeding with definition of the methodology for the consumer loans. In the context of this path, a product can be a node or a leaf in the product hierarchy, or a standalone leaf when no hierarchies are associated with the rule.

Once you have created a Transfer Pricing rule and a version, you can assign transfer pricing methodologies to product-currency combinations in either of the following two ways:

- By creating a conditional assumption using conditional logic. See:
 - Associating Conditional Assumptions with Transfer Pricing Rules, page 2-21.
 - Overview of Conditional Assumptions, page 11-1.
 - Creating Conditional Assumptions, page 11-1.
- Directly on the Transfer Pricing methodology page, as described here.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing rule, page 12-2

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Yield Curve Term	Defines the point on the yield curve that the system references to calculate transfer rates.
Historical Term	Specifies the period over which the average is to be taken for the Moving Averages method, and yield curve from a date earlier than the Assignment Date for the Spread from Interest Rate Code method.
Lag Term	Specifies a yield curve from a date earlier than the Assignment Date for the Spread from Interest Rate Code method.
Rate Spread	The fixed positive or negative spread from an Interest Rate Code or Note Rate, used to generate transfer rates in the Spread from Interest Rate and Spread from Note Rate methods.

Term	Description
Model with Gross Rates	This option becomes available when you select Account tables as the data source and allows you to specify whether modeling should be done using the net or gross interest rate on the instrument. This option is only applicable when the Net Margin Code is also set to one, for example, Fixed. Gross rates are typically selected while modeling the effect of serviced portfolios where the underlying assets have been sold but the organization continues to earn servicing revenue based on the original portfolio.
Mid Period	This option applies to adjustable rate instruments only. It dictates whether the transfer rate is based on the last repricing date, current repricing period, prior repricing date, or some combination thereof.
Assignment Date	This is the effective date of the yield curve.
Percentage/Term Points	The term points that the system uses to compute the Redemption Curve method results. A percentage determines the weight assigned to each term point when generating results.
Add Dimension Values	Allows you to select the products that you want to transfer price using the Unpriced Account method.
Across All Organizational Units	When this option is enabled, transfer price is calculated as a weighted average across all Organizational Units for the matching product value. Otherwise, transfer price is calculated from accounts only within a particular Organizational Unit.

1. Navigate to the version attributes page.
2. Select a currency or product from the list of values, depending on whether you want to define transfer Pricing methodologies for the entire product portfolio, one currency at a time; or for all the currencies supported in your system, one product at a time.

3. Click Go. The system displays a list of all the products (for which you can define assumptions) or currencies (that are active in the system).
4. Click Update corresponding to the product-currency combination for which you want to define the Transfer Pricing Methodology.
5. Select the appropriate data source: Account Tables or Management Ledger Table.
6. Select the required Transfer Pricing method.

Important: If this field is left blank, transfer prices will not be generated for the concerned product-currency combination. The Transfer Pricing methodologies available depend on the selected data source. See: Successful Transfer Pricing Combinations, page 12-7.

Depending on the transfer pricing method selected, certain required and optional parameter fields are displayed. You can update these fields as required. See: Required Parameters for a Transfer Pricing Methodology, page 12-8. See also:

- Defining the Redemption Curve methodology, page 12-9.
 - Defining the Unpriced Account Methodology, page 12-10.
7. Specify the desired Option Cost methodology. This option is available only when the data source is Account Tables. Here is how you can go about specifying an Option Cost Methodology:
 1. Select Run using Monte Carlo Option Cost Method. The Target Balance drop-down list is displayed.
 2. Select the required balance type. You can select any one of the following as the designated target balance for option cost calculations:
 - Par Balance
 - Book Balance
 - Market Value

See: Transfer Pricing Option Cost, *Oracle Financial Services Reference Guide* and Transfer Pricing Process Rule and Option Costs, page 2-37.

8. Click Apply.
The version attributes page is displayed.

At this point you can:

- Continue defining additional methodologies for other product-currency combinations by repeating the above procedure.
- Complete the process by clicking Finish in the Create Process Flow or Apply in the Update Process Flow.

The new assumptions are saved and the Transfer Pricing rule selector page is displayed.

Guidelines

Availability of Transfer Pricing Methodologies

The availability of transfer pricing methodologies depends on the data source that you select: Account Table or Management Ledger Table. The following table describes the Transfer Pricing Methodologies available for each of these data sources and displays whether that methodology requires the selection of a Transfer Pricing Interest Rate Code.

Successful Transfer Pricing Combinations

Transfer Pricing Methodology	Data Source: Account Table	Data Source: Ledger Table	Interest Rate Code
None	Yes	Yes	
Cash Flow Weighted Term	Yes		Yes
Cash Flow Duration	Yes		Yes
Cash Flow Zero Discount Factors	Yes		Yes
Moving Averages	Yes	Yes	Yes
Straight Term	Yes		Yes
Spread from Interest Rate Code	Yes	Yes	Yes
Spread from Note Rate	Yes		

Transfer Pricing Methodology	Data Source: Account Table	Data Source: Ledger Table	Interest Rate Code
Redemption Curve	Yes	Yes	Yes
Unpriced Account		Yes	

Required Parameters

You cannot define a transfer pricing methodology successfully, unless you specify the required parameters. The following table displays the parameters associated with each transfer pricing method and specifies whether they are required or optional. The optional parameter fields display default values. However, you may decide to change the values for the optional parameters for certain methodologies, such as, the Redemption Curve or the Unpriced Account methods.

Required Parameters for a Transfer Pricing Methodology

Transfer Price Method	Yield Curve Term	Historical Term	Lag Term	Rate Spread	Assignment Date	Mid Period	Term Points	Dimension Values
Cash Flow Weighted Term								
Cash Flow Duration								
Cash Flow Zero Discount Factors								
Moving Averages	Optional	Optional						
Straight Term						Optional		
Spread from IRC	Optional		Optional	Optional	Optional	Optional		

Transfer Price Method	Yield Curve Term	Historical Term	Lag Term	Rate Spread	Assignment Date	Mid Period	Term Points	Dimension Values
Spread from Note Rate				Optional		Optional		
Redemption Curve					Optional	Optional	Required	
Unpriced Account								Required

Related Topics

- Setting Transfer Pricing Rules, page 2-3
- Overview of Transfer Pricing Rules, page 12-1
- Standard Navigation Paths, page A-1
- Creating Conditional Assumptions, page 11-1
- Defining the Redemption Curve Methodology, page 12-9
- Defining the Unpriced Account Methodology, page 12-10

Defining the Redemption Curve Methodology

As part of the process for defining the Redemption Curve methodology, you must select as many Term Points from the Transfer Pricing Yield curve as you desire and allocate the percentage runoff for each of those points.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing rule, page 12-2

Procedure to Add Term Points:

The steps involved in adding Term Points are listed below:

1. Click Select Term Points.
The Select Term Points page is displayed.
2. Select the Transfer Pricing Yield Curve Points as required.

3. Click Apply.

The Transfer Pricing methodology page is displayed.

4. Update the percentage value for each Term Point.

Note: The sum of all the percentages for all Term Points must add up to 100. To remove a Yield Curve Point from the Percentages/Term Points table, click the corresponding Delete icon for the Yield Curve Point.

Related Topics

Defining Transfer Pricing Methodologies, page 12-3

Standard Navigation Paths, page A-1

Defining the Unpriced Account Methodology

When defining an Unpriced Account methodology, you need to select the Line Item dimension members (products) whose weighted average transfer rate will be assigned to the product being defined.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing rule, page 12-2

Procedure to Add Dimension Values:

The steps involved in adding Dimension Values are listed below:

1. Click Add Dimensions.

The Add Dimensions page is displayed.

2. Search and select the required Line Item dimension members. Specify whether weighted average of transfer rates has to be taken across all organizational units or for accounts only within that organizational unit.

3. Click Apply.

The Transfer Pricing methodology page is displayed.

Related Topics

Defining Transfer Pricing Methodologies, page 12-3

Standard Navigation Paths, page A-1

Prepayment Rules

This chapter describes the procedure for working with and managing Prepayment rules.

This chapter covers the following topics:

- Overview of Prepayment Rules
- Creating Prepayment Rules
- Defining Prepayment Methodologies

Overview of Prepayment Rules

Prepayment rules allow you to specify methodologies to model the prepayment behavior of products in your portfolio and quantify the associated prepayment risk in monetary terms. See: *Setting Prepayment Rules*, page 2-27.

The methodologies contained in the Prepayment rule are referenced by the Transfer Pricing Process rule. These prepayment methodologies are used in combination with cash flow based transfer pricing methods to generate transfer pricing results.

The procedure for working with and managing the Prepayment rule is similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Prepayment rules. See: *Searching for Rules*, page 3-5.
- Creating Prepayment Rules, page 13-2.
- Viewing and Updating Prepayment rules. See: *Viewing and Updating Rules*, page 3-7.
- Duplicating Prepayment rules. See: *Duplicating Rules*, page 3-7.
- Deleting Prepayment rules. See: *Deleting Rules*, page 3-9.

As part of creating and updating Prepayment rules, you can also define prepayment

methodologies. See:

- Defining Prepayment Methodologies, page 13-3.
- Defining the Constant Prepayment Method, page 13-6.
- Defining the Prepayment Table Method, page 13-8.
- Defining the Arctangent Calculation Method, page 13-9.

Related Topics

Standard Navigation Paths, page A-1

Creating Prepayment Rules

You create a Prepayment rule to define prepayment assumptions for new products.

Procedure:

1. Navigate to the Prepayment rule home page.
2. Complete standard steps for this procedure. See: *Creating Rules*, page 3-6.

Important: In addition to the standard steps for creating rules, the procedure for creating Prepayment rule and version involves one extra step. After Standard Step 5, you can select a product hierarchy. You can define methodologies at any level of the hierarchical product dimension. The hierarchical relationship between the nodes allows inheritance of methodologies from parent nodes to children nodes.

You can also continue defining a methodology at the lowest level of the Line Item dimension without selecting a hierarchy.

Caution: Oracle Transfer Pricing does not support multiple value set hierarchies and hierarchies with multiple tops. In addition, all the hierarchies need to be stored in flattened format for processing by the application. See: *Working with Hierarchies, Oracle Enterprise Performance Foundation User's Guide*.

Related Topics

Overview of Prepayment Rules, page 13-1

Standard Navigation Paths, page A-1

Defining Prepayment Methodologies

The definition of prepayment methodologies is part of the Create or Update Prepayment rule process where prepayment assumptions are made for product-currency combinations. When you click Apply in the Create Prepayment rules process, the rule and the version are saved and the Prepayment rule home page is displayed. However, the prepayment methodology has not yet been defined for any of your products at this point. Typically, you would start defining your prepayment methodologies for product-currency combinations before clicking Apply.

The Prepayment rule supports definition of prepayment assumptions for combinations of two dimensions: Product and Currency. Consequently, before you start defining the prepayment methodologies for your products, you need to decide the way you want to input the parameters. You can select any one of the following two paths:

- All Products for one Currency
- All Currencies for one Product

Once you have created a Prepayment rule and a version, you can assign prepayment methodologies to product-currency combinations in any of the following two ways:

- By creating a conditional assumption using conditional logic. See:
 - [Associating Node Level and Conditional Assumptions with Prepayment Rules](#), page 2-35.
 - [Overview of Conditional Assumptions](#), page 11-1.
 - [Creating Conditional Assumptions](#), page 11-1.
- Directly on the Prepayment methodology page, as described here.

Prerequisites

- [Performing basic steps for creating or updating a Prepayment rule](#), page 13-2

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Calculation Method	The method used to model prepayment behavior of instruments. Oracle Transfer Pricing provides three prepayment calculation methods: Constant, Prepayment Table, and Arctangent.
Cash Flow Treatment	<p>Allows you to specify one of the following two ways in which prepayments are made for a particular product or product hierarchy:</p> <ul style="list-style-type: none">• Refinance: This is the most commonly used option. Select refinance to keep payment amounts after prepayment consistent with a portfolio-based assumption. This reduces the scheduled payment amount on each loan and maintains the same maturity term.• Curtailment: Select curtailment to change the periodic payment amounts due. The prepayments are treated as accelerated payments, with a payoff earlier than the originally scheduled term.
Market Rate	The market rate is defined as the sum of the Index (the yield curve rate as described by the Interest Rate Code) and the Spread (the difference between the customer rate and market rate).
Associated Term	<p>Allows you to define the term for the point on the yield curve selected in the Market Rate Definition that will be used in obtaining the market rate.</p> <ul style="list-style-type: none">• Remaining Term: The number of months remaining until the instrument matures.• Reprice Frequency: The frequency with which the instrument reprices. This defaults to the original term for a fixed rate instrument.• Original Term: The number of months that was the originally scheduled life of the instrument.

Term	Description
Prepayment Rate Definition	This table allows you to specify the constant annual prepayment rate that you want to apply to instruments having origination dates in a particular date range.
Seasonality	<p>This table allows you to specify seasonality adjustments. Seasonality refers to changes in prepayments that occur predictably at given times of the year.</p> <p>Seasonality adjustments are based on financial histories and experiences, and should be modeled when you expect the amount of prepayments made for certain types of instruments to increase or decrease in certain months.</p> <p>The default value for seasonality factors is 1, which indicates that no seasonality adjustment is made for a month. Changing the seasonality factors is optional. You can change the seasonality factors for none, one, or multiple months.</p> <p>To make seasonality adjustments, you need to enter a value between 0.00 and 99.99 for the seasonality factors associated with each month. Seasonality factors less than 1 mean that prepayments are decreased for a particular month. Seasonality factors greater than 1 indicate that prepayments are increased for a particular month.</p>

1. Navigate to the Prepayment methodology page.
2. Select a Calculation Method, Constant, Prepayment Table, or Arctangent, and click Go.

Note: The default value for the Calculation Method drop down list is None. If you select None as the calculation method, no prepayment assumptions will be assigned to the particular product-currency combination.

3. Select a Cash Flow Treatment type, Refinance or Curtailment, and click Go to refresh the screen with the relevant parameters.

Note: Refinance is the most commonly used method.

4. Define the parameters and annual prepayment rates for the selected calculation method: Constant, Prepayment Table, or Arctangent.

Important: The parameters displayed on the Prepayment methodology page vary depending on the calculation method (Constant, Prepayment Table, or Arctangent) that you have selected. See:

- Defining the Constant Prepayment Method, page 13-6.
- Defining the Prepayment Table Method, page 13-8.
- Defining the Arctangent Calculation Method, page 13-9.

5. Click Apply.

The Prepayment version definition page is displayed.

At this point you can:

- Continue defining additional methodologies for other product-currency combinations by repeating the above procedure.
- Complete the process by clicking Finish in the Create Process Flow or Apply in the Update Process Flow.

Note: When you click Finish, the prepayment assumptions are saved and the Prepayment rule home page is displayed.

Related Topics

Prepayment Methodologies and Rules, page 2-27

Creating Conditional Assumptions, page 11-1

Overview of Prepayment Rules, page 13-1

Standard Navigation Paths, page A-1

Defining the Constant Prepayment Method

Use this procedure to define prepayment assumptions using the Constant Prepayment method.

Prerequisites

- Performing basic steps for creating or updating a Prepayment rule, page 13-2

Procedure:

1. Select the Start Origination Date using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.

Note: The first cell in the Start Origination Date column and all of the cells in the End Origination Date column are read only. This ensures that all the possible origination dates are included in cash flow calculation runs. Each row in the End Origination Date column is filled in by the system when a new Start Origination Date is entered into the next row.

The first Start Origination Date (in row 1) has a default value of January 1, 1900. When you enter a Start Origination Date in the next row, the system inserts a date that is a day prior to the previous End Origination Date field.

2. Enter the annual prepayment rate percent that you want to apply to the instruments having origination dates in a particular Start Origination-End Origination Date range.

Note: The Percent column represents the actual annualized prepayment percentage that the system uses to generate the principal runoff during the cash flow calculations.

3. Click Add Another Row to add additional rows and click the corresponding Delete icon to delete a row.

You can add as many rows in this table as you require. However you need to enter relevant parameters for each new row.

4. Define the Seasonality as required.

Related Topics

Constant Prepayment Method, page 2-28

Defining Prepayment Methodologies, page 13-3

Standard Navigation Paths, page A-1

Defining the Prepayment Table Method

Use this procedure to define prepayment assumptions using the Prepayment Table Calculation method.

Prerequisites

- Performing basic steps for creating or updating a Prepayment rule, page 13-2

Procedure:

1. Select an Index from the list of values.
2. Enter the Spread.
A Spread is the difference between the Customer Rate and the Market Rate.
3. Select an Associated Term: Remaining Term, Reprice Frequency, or Original Term.
4. Specify the Prepayment Table parameters.
 1. Select the Start Origination Date using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.
 2. Enter the Coefficient by which the Prepayment Rate should be multiplied.
This multiple is applied only to the instruments for which the origination date lies in the range defined in the Start Origination Date-End Origination Date fields.
 3. Select a predefined prepayment table from the Prepayment Table Rule list of values.
The system uses the prepayment table assumptions to calculate the prepayment amounts for each period. You need to associate a prepayment table for every Start Origination-End Origination Date range.
 4. Click Add Another Row to add additional rows and click the corresponding Delete to delete a row.
You can add as many rows in this table as you require. However you need to enter relevant parameters for each new row.
5. Define the Seasonality as required.

Related Topics

Prepayment Table Method, page 2-28

Prepayment Table Rules, page 14-1

Defining Prepayment Methodologies, page 13-3

Standard Navigation Paths, page A-1

Defining the Arctangent Calculation Method

Use this procedure to define prepayment assumptions using the Arctangent Calculation method.

Prerequisites

- Performing basic steps for creating or updating a Prepayment rule, page 13-2

Procedure:

1. Select an Index from the list of values.
2. Enter the Spread.
A Spread is the difference between the Customer Rate and the Market Rate.
3. Select an Associated Term: Original Term, Reprice Frequency, or Remaining Term.
4. Specify the Arctangent Argument table parameters.
 1. Select the Start Origination Date using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.
 2. Enter the values for the Arctangent parameters (columns K1 through K4) for each Start Origination Date in the table. The valid range for each parameter is -99.9999 to 99.9999.
 3. Click Add Another Row to add additional rows and click the corresponding Delete to delete a row.
You can add as many rows in this table as you require. However you need to enter relevant parameters for each new row.
5. Define the Seasonality as required.

Related Topics

Arctangent Calculation Method, page 2-31

Defining Prepayment Methodologies, page 13-3

Standard Navigation Paths, page A-1

Prepayment Table Rules

This chapter describes the procedure to build prepayment tables using Prepayment Table Rules.

This chapter covers the following topics:

- Overview of Prepayment Table Rules
- Creating Prepayment Table Rules
- Updating Prepayment Tables
- Updating Prepayment Rates in a Prepayment Table

Overview of Prepayment Table Rules

The Prepayment Table rule allows you to build prepayment tables. These prepayment tables can be referenced by a Prepayment Rule to model prepayment behavior of instruments. See: Prepayment Table Method, page 2-28 and Prepayment Rules, page 13-1.

The procedure for working with and managing the Prepayment Table rule is similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Prepayment Table rules. See: Searching for Rules, page 3-5.
- Creating Prepayment Table Rules, page 14-2.
- Viewing and Updating Prepayment Table rules. See:
 - Viewing and Updating Rules, page 3-7.
 - Updating Prepayment Tables, page 14-6.
- Duplicating Prepayment Table rules. See: Duplicating Rules, page 3-7.
- Deleting Prepayment Table rules. See: Deleting Rules, page 3-9.

Related Topics

Standard Navigation Paths, page A-1

Creating Prepayment Table Rules

Creating a Prepayment Table rule comprises the following sub procedures:

- Creating Prepayment Table rule and version, page 14-2
- Defining the structure of the prepayment table, page 14-4
- Assigning Node Values, page 14-5

Procedure to create a Prepayment Table Rule and Version:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Dimension	Influences the prepayment behavior of an instrument. You can build a prepayment table using up to three prepayment dimensions. Each dimension maps to an attribute of the underlying transaction (age/term or rate) so the cash flow engine can apply a different prepayment rate based on the specific characteristics of the record.

Term	Description
Lookup method	<p>Used to calculate prepayment rates for the prepayment dimension values that do not fall on the defined prepayment dimension nodes. Oracle Transfer Pricing offers the following look up methods:</p> <ul style="list-style-type: none"> <li data-bbox="787 457 1458 617">• Interpolation: Under this method, the prepayment rates are determined by calculating an exact value on an axis. This method assumes that prepayment speeds change on a straight-line basis between the two nodes and calculates accordingly. <li data-bbox="787 655 1458 781">• Range: Under this method, the prepayment rates are determined by calculating a range of values on an axis. This method assumes that the prepayment speed will remain the same for the entire range. <p>The following example explains the differences between these two look up methods. The following lists show the age and corresponding prepayment rates of instruments.</p> <p>Age</p> <ul style="list-style-type: none"> <li data-bbox="787 974 862 1001">• 12 <li data-bbox="787 1039 862 1066">• 24 <li data-bbox="787 1104 862 1131">• 36 <li data-bbox="787 1169 862 1197">• 60 <p>Prepayment Rates</p> <ul style="list-style-type: none"> <li data-bbox="787 1283 846 1310">• 5 <li data-bbox="787 1348 862 1375">• 10 <li data-bbox="787 1413 862 1440">• 15 <li data-bbox="787 1478 862 1505">• 20 <p>Under the Interpolation method, the prepayment speeds increase gradually. In this example, the Interpolated prepayment rate of an instrument aged 30 months is 12.5%. This is exactly halfway between the 10% and 15% rate. However, under the Range method, the Prepayment speeds increase in steps. Using the Range method, the prepayment rate is 10%, as this rate percentage would apply to the range from</p>

Term	Description
	24 months to 35.9999 months.
Nodes	Exact points for each dimension where attribute information has been defined.

1. Navigate to the Prepayment Table Rule home page.
2. Complete standard steps for this procedure. See: Creating Rules, page 3-6.

Procedure to Define the Structure of the Prepayment Table

When you click Continue after defining a Prepayment Table version, the Define Dimensions page is displayed. The Prepayment Table consists of the Prepayment Dimensions and the Node Values for these Dimensions which you select on this page. To model the prepayment behavior, you can select a maximum of three prepayment dimensions. Once the dimensions and the number of nodes are defined, you need to assign values to the nodes.

Note: You can use the analogy of a three dimensional table to understand how to deal with the prepayment dimensions. The first dimension you select would resemble the row (X-axis). The second dimension would act as the column (Y-axis). The final third dimension will be the page (Z-axis).

3. Select the first Dimension.
4. Select a lookup method for that Dimension.
5. Enter the number of Nodes for the Dimension.
This number may vary from dimension to dimension.
6. If required, repeat the previous three steps for up to two additional Dimensions.

Important: There are certain restrictions while defining Dimensions:

- You must select the Dimension type for a row and define the values for that dimension.
- You cannot define the second (row) dimension until you have defined the first (row) dimension. Similarly, the third

dimension can not be defined until you have defined the first two dimensions.

7. Click Go.

The Define Dimensions page is refreshed. You can now assign the Node Values for each dimension. At this point, you can also modify the structure of the table, if required.

Modifying the Table Structure

- To add more Nodes to a particular Dimension, update the number of Nodes for the Dimension and click Go.
- To delete Nodes from a particular Dimension, click the delete icon corresponding to the Node.

Note: Nodes cannot be deleted by reducing their numbers. Also, all Nodes cannot be deleted and at least one Node must exist in each Dimension.

To change the method of a particular Dimension, select the required method from the corresponding list of methods.

Procedure to Assign Node Values

8. Assign values for each of the Nodes.

9. Click Apply.

The Rule, Version, Prepayment Dimensions, and Nodes are saved. You cannot change the Dimensions once the Prepayment Table rule has been saved.

Note: At this point, only the structure of the Prepayment Table has been defined. The Prepayment Speeds for all the Nodes still need to be defined. See: Adding Prepayment Speeds to the Prepayment Table, page 14-7.

10. Enter the Prepayment Speeds in the Prepayment Table.

Node values for the row and column dimensions are displayed as a table, while the node values for the page dimensions (if selected) are shown in the drop down list.

11. Repeat the process for all node values of the page driver. To change the node value along the page driver, select the required node value from the drop-down list.

Note: Node values will be displayed in the drop-down list only if you selected three drivers.

12. Click Apply. The Prepayment Rates are saved and the Prepayment Table Rule home page is displayed.

Related Topics

Prepayment Table Method, page 2-28

Standard Navigation Paths, page A-1

Overview of Prepayment Table Rules, page 14-1

Updating Prepayment Tables

As part of updating Prepayment rules, you can modify Prepayment speeds and the structure of the Prepayment Table to a certain extent. You can also modify the lookup methods (Range or Interpolation) used for calculating, the number of Nodes, and the actual values of the Nodes. However, you cannot update the dimensions.

Prerequisites

- Predefined Prepayment Table rules

Procedure:

1. Search for the Prepayment Table rule, which you want to update. See: Searching for Rules, page 3-5.
2. Click Update corresponding to the required version.
The Update Prepayment Table version page is displayed.

Procedure to Update Speeds

1. Modify the Prepayment Speeds in the table as required. See Updating Prepayment Rates in a Prepayment Table, page 14-7.
3. Click Apply.
The Prepayment Table rule home page is displayed.

Related Topics

Prepayment Table Method, page 2-28

Standard Navigation Paths, page A-1

Overview of Prepayment Table Rules, page 14-1

Updating Prepayment Rates in a Prepayment Table

Once the basic structure of the prepayment table has been created, prepayment rates can be added to, or modified for, each of the node values along the chosen dimensions. Use this procedure to add or update annual prepayment rates in the prepayment table.

Procedure:

1. Search for the Prepayment Table rule, for which you want define prepayment rates. See: Searching for Rules, page 3-5.

2. Click Update corresponding to the required version.

The Update Prepayment Table version page is displayed.

3. Enter the Prepayment Speeds in the Prepayment Table.

Node values for the row and column dimension are displayed as a table on the Update Prepayment Table version page, while the node values for page dimension (if selected) are shown in the drop down list.

4. Repeat the process for all node values of the page dimension. To change the node value along the page dimension, select the required node value from the drop-down list.

Note: Node values will be displayed in the drop-down list only if you selected three dimensions.

5. Click Apply.

The table with updated prepayment rates is saved and the Prepayment Table home page is displayed.

Related Topics

Prepayment Table Method, page 2-28

Standard Navigation Paths, page A-1

Overview of Prepayment Table Rules, page 14-1

Propagation Pattern

This chapter describes the procedure for defining the propagation pattern.

This chapter covers the following topics:

- Overview of the Propagation Pattern
- Defining the Propagation Pattern
- Propagating Transfer Pricing Results

Overview of the Propagation Pattern

The Propagation Pattern allows you to propagate transfer rates and option costs for any applicable instrument table from a prior period. See:

- Defining the Propagation Pattern, page 15-1.
- Propagating Transfer Pricing Results, page 15-3.
- Transfer Pricing Process Rule and Propagation Patterns, page 2-39.

Related Topics

Standard Navigation Paths, page A-1

Defining the Propagation Pattern

Use this procedure to define the Propagation pattern.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Processing Table	Account tables that have been enabled for transfer pricing or option cost processing. These tables are sorted alphabetically.
Source Table	Tables that are referenced to obtain the previously calculated transfer rates or option costs.
Historical Lag	The time difference between the transfer rate or option cost processing runs of the source and processing tables.
Frequency	A numeric value multiplied with a Multiplier to calculate the Historical Lag.
Multiplier	The unit value of the Frequency.

1. Navigate to the Propagation Pattern home page.
2. Select the Source Table that needs to be associated with the Processing Table.

Note: The Source Table for any propagation process can be either the same table (if you store multiple periods of instrument data in the same Account table) or a separate table (if you store historical records in separate Account tables).

3. Specify the historical lag between the processing and source tables.
 1. Select the Frequency.
 2. Select the Multiplier.

Note: The prior period source date for each Source Table is defined in relation to the current effective date. For instance, if you transfer price on a monthly basis, you should specify the historical lag between the processing and source tables as one month.

4. Click Apply.

The Propagation Pattern that you defined is saved.

Related Topics

Overview of the Propagation Pattern, page 15-1

Standard Navigation Paths, page A-1

Propagating Transfer Pricing Results

Depending upon your requirements, you can choose to propagate either the Transfer Rate information or the Option Cost information or both.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing Process rule, page 16-2.

Procedure:

1. Navigate to the Transfer Pricing Process run page.
2. Select the propagation parameters:
 - **Transfer Rates:** Selecting Transfer Rates updates all term-related instrument records which have instrument-level history for a prior period with the transfer rate that applied in that prior period.
 - **Option Costs:** Selecting Option Costs updates all term-related instrument records which have instrument-level history for a prior period with the option cost data that applied in that prior period.

See: Transfer Pricing Process Rules, page 16-1.

Note: When a table is updated using a propagation pattern, an instrument record must satisfy the following criteria to receive a transfer rate or option cost.

- It must be an instrument that exists in both the Target (processing) table (with the current effective date) and the Source table (with the prior period effective date).
- The instrument must also satisfy one of the following conditions:

- It must be a fixed-rate (Repricing Freq = 0 in Target table) instrument.
- It must be an adjustable-rate (Repricing Freq \neq 0 in Target table) instrument with Target Last Repricing Date \leq Prior Period Effective Date. In other words, it must be an adjustable-rate instrument that has not repriced since the prior period.

The matched spread is migrated from the prior period record and not recomputed from the Transfer Rate and Current Rate on the target table record.

Related Topics

[Overview of the Propagation Pattern, page 15-1](#)

[Standard Navigation Paths, page A-1](#)

Transfer Pricing Process Rules

This chapter discusses the procedure for working with and managing the Transfer Pricing Process rules.

This chapter covers the following topics:

- Overview of Transfer Pricing Process Rules
- Creating a Transfer Pricing Process Rule
- Executing a Transfer Pricing Process Rule

Overview of Transfer Pricing Process Rules

The Transfer Pricing Process rule allows you to perform the following tasks:

- Determine the data that you want to process.
- Submit to the Transfer Pricing engine the transfer pricing and prepayment assumptions you want to process.
- Specify to the Transfer Pricing engine whether you want to generate transfer rates or option costs, or both.
- Specify to the Transfer Pricing engine whether you want to calculate or propagate transfer pricing results.
- Calculate and migrate charges and credits for funds provided or used for a combination of dimensions to the Management Ledger table (FEM_BALANCES).
- Formulate and execute the transfer pricing request and generate results.

See: Setting Transfer Pricing Process Rule, page 2-36.

The procedure for working with and managing the Transfer Pricing Process rule is similar to that of other Oracle Transfer Pricing business rules. It includes the following steps:

- Searching for Transfer Pricing Process rules. See: Searching for Rules, page 3-5.
- Creating Transfer Pricing Process Rules, page 16-2.
- Viewing and Updating Transfer Pricing Process rules. See: Viewing and Updating Rules, page 3-7.
- Duplicating Transfer Pricing Process rules. See: Duplicating Rules, page 3-7.
- Deleting Transfer Pricing Process rules. See: Deleting Rules, page 3-9.

The Transfer Pricing Process rule has a separate Run page. The Run page allows you to execute a Transfer Pricing Process rule. See: Executing a Transfer Pricing Process Rule, page 16-9.

Related Topics

Standard Navigation Paths, page A-1

Creating a Transfer Pricing Process Rule

You create a Transfer Pricing Process rule to formulate and execute transfer pricing or option cost processing requests.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing rule, page 12-2.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Transfer Pricing rule	This LOV allows you to select a Transfer Pricing rule, a Transfer Rate parameter. The Transfer Pricing rule provides the application with transfer pricing assumptions that you want to apply to products to calculate their transfer rates.

Term	Description
Transfer Pricing Rule Product Hierarchy	Displays the product hierarchy associated with the selected Transfer Pricing rule. A read only field, it gets populated automatically when you select a Transfer Pricing rule. Transfer Pricing Rule Product Hierarchy is a read only Transfer Pricing parameter.
Prepayment Rule	This LOV allows you to select a Prepayment rule, a Transfer Pricing parameter. The Prepayment rule provides the application with prepayment assumptions that you want to apply to products in conjunction with cash flow transfer pricing methods.
Prepayment Rule Hierarchy	Displays the product hierarchy associated with the selected Prepayment rule. A read only field, it gets populated automatically when you select a Prepayment rule. Prepayment Rule Hierarchy is a Transfer Pricing parameter.
Term Structure Model	An Option Cost parameter, it governs the generation of one-month stochastic rates, discount factors for each scenario, and discrete rates for any maturity used in calculating the option-adjusted spread.
Smoothing Method	An Option Cost parameter, it is used to interpolate rates on the valuation curve for terms that fall between given points.
Rate Index Rule	An Option Cost parameter, it is used to define the rates used to index adjustable-rate Instruments under the different Rate Paths generated by stochastic processing. The rates are defined automatically in terms of the valuation curve.
Number of Rate Paths	An Option Cost parameter, it ranges between 1 and 2000. Greater numbers of rate paths increase accuracy but also increase processing time. Experiment to find the optimal level for your institution's portfolio.

Term	Description
Random Number Generation Method	<p data-bbox="781 306 1317 432">An Option Cost parameter, it determines how the Monte Carlo process selects random numbers. The Random Number Generation Method has two variations:</p> <ul data-bbox="781 457 1365 848" style="list-style-type: none"><li data-bbox="781 457 1365 653">• Low Discrepancy Sequences: Low-discrepancy sequences, also known as quasirandom sequences, are designed to fill the space uniformly. These achieve better accuracy than pseudorandom sequences when applied to numerical problems, integration in high dimension, and so on.<li data-bbox="781 688 1365 848">• Pseudorandom Sequences: These are the traditional random numbers generated by most compilers. They are designed to do well on some statistical tests: low auto correlation, high period before the sequence repeats itself.

Term	Description
Charge/Credit Accrual Basis	<p>A migration parameter, Charge/Credit Accrual Basis denotes the basis on which the interest accrual is calculated. You need to select the accrual factor to be applied when calculating the cost of funds.</p> <p>The cost of funds is calculated using the following formula:</p> <p>Cost of Funds = Balances x Assigned Transfer Rate x Charge/Credit Accrual Factor</p> <p>Oracle Transfer Pricing offers the following accrual basis options:</p> <ul style="list-style-type: none"> • 30/360: This is the default Charge/Credit Accrual Basis option. It applies the accrual basis calculation of 30 days divided by 360 days. • Actual/360: Applies the accrual basis calculation of number of days in the month divided by 360 days. • Actual/Actual: Applies the accrual basis calculation of number of days in the month divided by number of days in the year. • 30/365: Applies the accrual basis calculation of 30 days divided by the 365 days. • 30/Actual: Applies the accrual basis calculation of 30 days divided by the number of days in the year. • Actual/365: Applies the accrual basis calculation of number of days in the month divided by 365 days.

Term	Description
Currency Output	<p>A migration parameter, it allows you to select the output currency. Oracle Transfer Pricing offers you the following currency output options:</p> <ul style="list-style-type: none"> • Entered and Functional Currency • Functional Currency Only <p>The following example illustrates the difference between entered and functional currency. A bank's loan may have Yen as entered currency. However, the bank might use US dollar to display its consolidated annual results. In this case, US dollar is the functional currency. In other words, the currency in which an organization keeps its books is its functional currency.</p>
Migration Dimensions	<p>Dimensions for which you want to migrate charges or credits, for funds provided or used, to the Management Ledger Table (FEM_BALANCES). Oracle Transfer Pricing provides you with a Shuttle Control component to specify migration dimensions.</p>
Available Dimensions	<p>One of the two Shuttle Control windows, it contains the names of the dimensions available for inclusion during the migration process. The dimensions available are:</p> <ul style="list-style-type: none"> • CHANNEL_ID • COMPANY_COST_CENTER_ORG_ID • NATURAL_ACCOUNT_ID • LINE_ITEM_ID • PRODUCT_ID <p>Important: In addition, you can define up to ten other dimensions.</p>

Term	Description
Selected Dimensions	<p>One of the two Shuttle Control windows, it contains the names of the dimensions that have already been selected for migration. While the LINE_ITEM_ID is a mandatory selected dimension, NATURAL_ACCOUNT_ID, and COMPANY_COST_CENTER_ORG_ID are default selected dimensions.</p> <p>Important: If required, you can deselect the default Natural Account and Company Cost Center Org dimensions to exclude them from the migration process.</p>

1. Navigate to the Transfer Pricing Process rule home page.
2. Complete standard steps for this procedure. See: Creating Rules, page 3-6.

Important: While creating Transfer Pricing Process rule and version, you need to perform the following subgroup of steps as part of the Standard Step 10.

Additional Steps to Specify Transfer Pricing Process Version Parameters

3. Select the Transfer Rate parameters:

- Transfer Pricing Rule

Important: You cannot create a Transfer Pricing Process rule without selecting a Transfer Pricing Rule.

- Prepayment Rule

4. Specify the Option Cost parameters:

- Term Structure Model
- Smoothing Method
- Rate Index Rule
- Number of Rate Paths

- Random Number Generation Method

5. Select the Migration parameters:

- Use Line Item Accrual Code option or Charge/Credit Accrual Basis global option

Tip: The Use Line Item Accrual Code option takes precedence over Charge/Credit Accrual Basis global option.

- Currency Output
- Migration Dimensions

Note: Oracle Transfer Pricing provides you with a Shuttle Control component to select the dimensions for the migration process. The two areas of the Shuttle Control are Available Dimensions and Selected Dimensions.

You can move dimensions from Available Dimensions into Selected Dimensions and vice versa by using the Move, Move All, Remove, and Remove All buttons. You can deselect the default Natural Account and Company Cost Center Org Dimensions if required, to exclude them from the migration process. For example, you may use one Transfer Pricing Process rule with the default dimensions included for normal production processing. A second Transfer Pricing Process rule can additionally be created to generate and migrate charges and credits for a product profitability view of results, using only the Line Item and Product dimensions.

When using Visual Trace to drill down from a Transfer Pricing charge or credit in the Transfer Pricing Process rule that created them, you will be able to review what dimension choices were used in the process.

Related Topics

Overview of Transfer Pricing Process Rules, page 16-1

Standard Navigation Paths, page A-1

Overview of the Oracle Transfer Pricing Process, page 2-1

Transfer Pricing Rules, page 12-1

Prepayment Rules, page 13-1

Executing a Transfer Pricing Process Rule

You execute a Transfer Pricing Process rule to generate transfer pricing or option cost results. Executing a Transfer Pricing Process rule involves specifying the parameters necessary for successfully running an active version of the rule, and running the version.

Important: Only a single version of a Process Rule can be executed. This version must be active in the production environment to generate production results.

Prerequisites

- Performing basic steps for creating or updating a Transfer Pricing Process rule, page 16-2.

Procedure:

This table describes some terms in the pages used for this procedure.

Selected Terminology

Term	Description
Propagation	<p>The propagation process copies historical results, either transfer rates or option costs, or both, that were generated by the application in a previous run for a prior period to the current period records. Some instruments do not require recalculation of transfer rates or options costs because these do not change for them from previously calculated results. For example:</p> <ul style="list-style-type: none">• The transfer rates of fixed-rate instruments (repricing frequency equals zero) do not change over the life of the instruments.• The transfer rates of some adjustable-rate instruments (repricing frequency greater than zero) may not change from the previous run if their last repricing date is less than or equal to the prior period effective date.

Term	Description
Skip Non-Zero Transfer Rate Records	A Transfer Pricing and Option Cost Calculation option. When it is used in conjunction with the Transfer Rate calculation type, transfer rates are generated only for records that do not have previous transfer rate information. All records that already have this information are skipped.
Skip Non-Zero Option Cost Records	A Transfer Pricing and Option Cost Calculation option. When it is used in conjunction with the Option Cost calculation type, option costs are generated only for records that do not have previous option cost information. All records that already have this information are skipped.
Calculation Mode	<p>You can calculate transfer rates or option costs, or both, in either of the following calculation modes:</p> <ul style="list-style-type: none"> <p>Standard: Select Standard calculation mode to calculate transfer rates for instrument records based on the Origination Date or Last Repricing Date of the instruments. It can also be used to calculate option costs based on the Origination Date.</p> <p>For all transfer pricing methods, Standard calculation mode writes instrument results to the Transfer Rate and Matched Spread columns.</p> <p>For all option cost calculation methods, Standard calculation mode writes instrument results to CUR_OAS and CUR_STATIC_SPREAD columns (Option Cost = Static Spread - Option Adjusted Spread).</p> <p>Remaining Term: Select Remaining Term calculation mode to calculate transfer rates and option costs for instrument records based on the remaining term of the instrument from the Effective Date (As of Date) of the data, rather than the Origination Date or Last Repricing Date of the instrument. This option treats your portfolio as if you acquired it on the Effective Date of your data.</p> <p>For all transfer pricing methods, Remaining Term Calculation Mode writes instrument results to TRAN_RATE_REM_TERM column and no matched spread is calculated.</p> <p>For all option cost calculation methods, Remaining Term calculation mode writes instrument results to HISTORIC_OAS and HISTORIC_STATIC_SPREAD columns (Option Cost = Static Spread - Option Adjusted Spread).</p>

Term	Description
Audit Options	<p>Oracle Transfer Pricing offers you the following audit options:</p> <ul style="list-style-type: none"> Detailed Cash Flows: Detailed Cash Flows record all events (Cash Flow and Repricings) that occur for five separate records (the first five records processed). Each of these records has all financial elements for all relevant cash flow dates populated in the FTP_PROCESS_CASH_FLOWS table. The data in this table uses OBJECT_ID as one of the main identifying keys. This table enables you to validate your transfer pricing results in a spreadsheet. Forward Rates: This option allows you to audit the static spread calculations by writing out calculated forward rates. 1 Month Rates: This option allows you to audit the option-adjusted spread calculations by writing out the different paths of one-month rates.
Ledger Migration	<p>The process in which the application generates charges or credits, for funds provided or used, for migration to the Management Ledger table based on the transfer rates or option costs obtained from Propagate or Calculate processing.</p>
Processing Tables	<p>Account tables that can be, or are already, included in a Transfer Pricing Process rule run for ledger migration. Oracle Transfer Pricing provides you with a Shuttle Control component to select the processing tables.</p>
Available Tables	<p>One of the two Shuttle Control windows, it displays the Account tables which you can include during a Transfer Pricing Process rule run for ledger migration. Only the account tables for which you have access privileges are displayed.</p>
Selected Tables	<p>One of the two Shuttle Control windows, it displays the tables that you select from the Available Tables. This window retains the names of the tables that you selected previously. For example, if you select two tables and save the Process Rule, the application will show these two tables the next time you open the rule.</p>
Condition	<p>Allows you to filter processing table records or omit certain tables.</p>

1. Navigate to the Transfer Pricing Process rule home page.
2. Search for the rule you want to execute, page 3-5.
3. Click Run corresponding to the version you want to execute. The Transfer Pricing

Process run page is displayed. The run page has the following six sections for which you can define parameters before executing the rule:

- Run time system parameters
- Propagation
- Transfer Rate and Option Cost Calculation
- Audit Options
- Ledger Migration
- Processing Tables

4. Select the Transfer Pricing Process run page parameters.

- Effective Date
- Calendar Period
- Ledger
- Dataset Group

Note: The application also displays in read only mode the names of the rules (Transfer Pricing, Prepayment, and Rate Index rules) that contain the assumptions you want to execute.

5. Select the Propagation parameters:

- **Transfer Rates:** Selecting Transfer Rates option updates all term-related instrument records which have instrument-level history for a prior period with the transfer rate that applied in that prior period.
- **Option Costs:** Selecting Option Costs updates all term-related instrument records which have instrument-level history for a prior period with the option cost data that applied in that prior period.

Important: You can choose to propagate either transfer rates or option costs, or both, or none.

6. Select the Transfer Rate and Option Cost Calculation parameters. You need to specify the following calculation parameters:

1. **Calculation Type:** You can calculate either Transfer Rates or Option Costs, or

both. You have the option of calculating transfer rates or option costs either for all records in the tables that satisfy the terms of the condition rule or skip nonzero records.

Tip: The option cost calculation related parameters are disabled, if the Transfer Pricing Process rule does not have a Rate Index rule associated with it.

2. **Calculation Mode:** You may select either the Standard or the Remaining Term calculation mode, as required.

Important: Even if you do not choose to calculate either transfer price or option cost, the choice of the calculation mode impacts the Ledger Migration process as follows:

- **Standard:** This leads to the migration of Transfer Rate and Historic Option Cost results.
- **Remaining Term:** This leads to the migration of Remaining Term Transfer Rate and Current Option Cost results.

7. Select the Audit Options. You may select some or all of the following audit options.

- Detailed Cash Flows

Note: The system ignores this option during processing in the absence of the transfer rate or option cost calculations parameters.

- Forward Rates
- 1 Month Rates

Tip: The Forward Rates and 1 Month Rates options apply only to Option Costs processing. These options are disabled if the Transfer Pricing Process rule does not have a Rate Index rule associated with it.

8. Select the Ledger Migration options. The available options are:

- None
- Transfer Rates

- Option Costs

9. Select the Processing Tables.

If required, you can apply a condition to omit certain tables or to filter records. You can either directly enter the name of the condition or search for it through the list of values provided and select it.

Note: Oracle Transfer Pricing provides you with a Shuttle Control component to select the processing tables. The two areas of the Shuttle Control are Available Tables and Selected Tables.

A table name shown in Selected Tables will not appear in Available Tables. Tables can be moved from Available Tables into Selected Tables and vice versa by using the Move, Move All, Remove, and Remove All buttons. You can also reorder the tables in each window.

10. Click Submit to execute (run) the rule. The Requests page is displayed.

Important: In case you do not want to run the rule immediately, click Apply to save the settings, and make a note of the Job ID displayed by the application. You can use the Job ID to schedule the execution of the rule on the Process Management tab. See: *About Concurrent Programs and Requests, Oracle Enterprise Performance Foundation User's Guide.*

Procedure to Determine the Object ID of the Transfer Pricing Process

11. Click Refresh repeatedly on the Requests page until the Phase (status) of the Transfer Pricing process is displayed as completed.

12. Click Output.

13. Note the Process ID, which is also the Object ID of the transfer pricing process.

You can use the Object ID to view the results of the transfer pricing process, such as detail cash flow audit results as follows:

- Creating a Condition associated with the Object ID Number. See: *About Conditions, Oracle Enterprise Performance Foundation User's Guide.*
- Querying the FTP_PROCESS_CASH_FLOWS table with a Data Inspector rule to view the data. See: *About the Data Inspector and Data Inspector Rules, Oracle Enterprise Performance Foundation User's Guide.*

Important: You can also use the Requests subtab on the Process Management tab to determine the Object ID of the transfer pricing process. See: *About Concurrent Programs and Requests, Oracle Enterprise Performance Foundation User's Guide*.

Related Topics

Overview of Transfer Pricing Process Rules, page 16-1

Standard Navigation Paths, page A-1

Overview of the Oracle Transfer Pricing Process, page 2-1

Transfer Pricing Rules, page 12-1

Prepayment Rules, page 13-1

Rate Index Rules, page 10-1

Propagation Patterns, page 15-1

Oracle Transfer Pricing Reports

This chapter describes the Oracle Transfer Pricing reports and the procedure for generating and viewing them.

This chapter covers the following topics:

- Overview of Oracle Transfer Pricing Reports
- Generating and Viewing Oracle Transfer Pricing Reports

Overview of Oracle Transfer Pricing Reports

Use Oracle Discoverer Plus 10gR2 to generate Oracle Transfer Pricing (FTP) reports and Enterprise Performance Foundations (FEM) data management reports, and to customize these reports in line with your reporting needs.

Oracle Discoverer Plus was chosen as the Oracle Financial Services (OFS) applications reporting solution because it is an Internet application tightly integrated with Oracle e-Business Suite Release 12. Discoverer Plus is a business intelligence solution that allows business users to retrieve and analyze data from Oracle databases by creating worksheets and charts, and to publish those results. See:

- Overview of the Oracle Financial Services Reporting Structure, *Oracle Financial Services Reporting Administration Guide*.
- *Oracle Business Intelligence Discoverer Plus User's Guide*.
- *Oracle Business Intelligence Discoverer Administration Guide*.
- *Oracle Business Intelligence Discoverer Viewer User's Guide*.

Oracle Transfer Pricing (FTP) reports leverage fact data defined in two core business areas, EPF and FTP, which refer to objects in the FEM schema. These business areas reside in a standard Oracle Applications end user layer (EUL_US) and are populated through facts, joins, and lookup tables.

The following table list the Oracle Transfer Pricing reports and their Discoverer file

names.

Oracle Transfer Pricing Reports

Report Name	Discoverer Workbook Title	Discoverer Worksheet Title
STD - FTP Interest Margin Account, page 17-5	STD - FTP Interest Margin Account Detail	FTP Interest Margin Account Detail
STD - FTP Interest Margin (Org/Product, Summary), page 17-6	STD - FTP Interest Margin (Org/Product Summary)	FTP Interest Margin (Org/Product Summary)
STD - FTP Margin Stratification, page 17-7	STD - FTP Margin Stratification	<ul style="list-style-type: none">• FTP Margin Stratification• FTP Margin Stratification Over Time

Oracle Transfer Pricing reports are crosstab reports. These reports:

- Include three query items: one on rows, one on columns, and one to serve as a measure or performance indicator defining what the data represents.
- Show data in columns and rows. However, the values at the intersection of rows and columns show summarized information rather than detailed information.
- Let you nest data to compare information using more than one query item in a column or in a row.

See:

- [Generating and Viewing Oracle Transfer Pricing Reports, page 17-2.](#)
- [Oracle Transfer Pricing Reports Common Concepts, page 17-3.](#)

Generating and Viewing Oracle Transfer Pricing Reports

Use the following procedure to generate and view Oracle Transfer Pricing (FTP) reports.

Procedure

1. Navigate to the Documents tab.
2. Click the report hyperlinks.

All the reports are displayed as hyperlinks under the sections Data Management Reports and Transfer Pricing reports. See:

- Oracle Transfer Pricing Reports Common Concepts, page 17-3.
- STD - FTP Interest Margin Account, page 17-5.
- STD - FTP Interest Margin (Org/Product, Summary), page 17-6.
- STD - FTP Margin Stratification, page 17-7.

Tip: Use the Back button of the browser to navigate back to the Transfer Pricing application. Open a new browser window to view both the Transfer Pricing and Discoverer Plus applications simultaneously.

Related Topics

[Overview of Oracle Transfer Pricing Reports, page 17-1](#)

[Oracle Transfer Pricing Reports Common Concepts, page 17-3](#)

[Standard Navigation Paths, page A-1](#)

Oracle Transfer Pricing Reports Common Concepts

The following concepts are common to all the Oracle Transfer Pricing reports.

Note: Report-specific information is provided in the report descriptions.

Business Area Folders

Folders setup in the Discoverer end user layer (EUL) or business area that contain database object, such as data and fact tables, on which Oracle Transfer Pricing reports are based.

Joins

A temporary relationship between two tables in a database query that lets you retrieve the exact data that you want. For example, the join *Line Items Dimension Hierarchy* -> *EPF Balances* is a one-to-many (1:n) join between *Line Items Dimension Hierarchy.Level20 id* and *EPF Balances.Line Item*.

The Oracle Transfer Pricing reports make use of the following joins to retrieve data:

- *Line Items Dimension Hierarchy* -> *EPF All Account Tables*

- EPF Datasets -> EPF All Account Tables
- EPF Ledgers -> EPF All Account Tables
- EPF Calendar Periods -> EPF All Account Tables
- EPF Currencies -> EPF All Account Tables

Page Items

Oracle Transfer Pricing reports are based on the following parameters:

- Calendar Period End Date
- Line Item Hierarchy Name
- Line Item Hierarchy Version
- Dataset
- Ledger
- Currency

Rows

The row dimension reflects the selected Line Item Hierarchy and Version in the two interest margin reports. The rows in the stratification report reflect the user defined ranges of transfer rate results.

Report Headings and Calculations (Columns)

The following headings and calculations are common to the Oracle Transfer Pricing reports:

- **Record Count:** COUNT(1)
- **Average Balance:** SUM(All Account Tables.Average Gross Book Balance)
- **Ending Balance:** SUM(All Account Tables.Current Gross Book Balance)
- **Current Rate:** SUM(EPF All Account Tables.Weighted Current Net Rate)/SUM(EPF All Account Tables."SUM(Current Gross Par Balance)")
- **Transfer Rate:** SUM(EPF All Account Tables.Weighted Transfer Rate)/SUM(EPF All Account Tables."SUM(Current Gross Par Balance)")
- **Matched Spread (%Margin):** SUM(EPF All Account Tables.Weighted Matched

Spread)/SUM(EPF All Account Tables."SUM(Current Gross Par Balance)")

- **Interest Income/Expense:** Current Net Rate*"SUM(Average Gross Book Balance) SUM"*(30/36000)
- **Transfer Charge/Credit:** Transfer Rate*"SUM(Average Gross Book Balance) SUM"*(30/36000)
- **Interest Margin:** "%Margin"*"SUM(Average Gross Book Balance) SUM"*(30/36000)

Related Topics

Overview of Oracle Transfer Pricing Reports, page 17-1

Generating and Viewing Oracle Transfer Pricing Reports , page 17-2

STD - FTP Interest Margin Account

This hierarchical report is based on account-level data and shows both the account level match funded spread and the interest margin for each product.

This reports lets you:

- Select the hierarchy and appropriate calendar period.
- Drill down through the various levels of the hierarchy.

Business Area Folders

The STD - FTP Interest Margin Account report is based on the following business area folders:

- **EPF All Account Tables:** This is a custom folder that joins instrument tables and presents views that link them.
- **Line Items Dimension Hierarchy:** This is a hierarchical dimension folder. Each hierarchy folder contains the hierarchy definition information and up to twenty levels of parent-child relationships. Within each level, the internal dimension member ID, member display code, and member name and description are available.

Rows

The STD - FTP Interest Margin Account report includes the following row:

- Line Item Hierarchy (up to 20 levels)

Related Topics

Overview of Oracle Transfer Pricing Reports, page 17-1

Oracle Transfer Pricing Reports Common Concepts, page 17-3

Generating and Viewing Oracle Transfer Pricing Reports , page 17-2

STD - FTP Interest Margin Account (Org/Product, Summary)

This hierarchical report is based on summarized ledger data and shows both the account-level match funded spread and the interest margin for each product.

Business Area Folders

The STD - FTP Interest Margin Account (Org/Product, Summary) report is based on the following business area folders:

- **EPF Balances:** This simple folder is based on the FEM_BALANCES table and contains summarized ledger data.
- **EPF Natural Account Dimension Hierarchy:** This is a hierarchical dimension folder based on the FEM_DIS_NAT_ACCTS_HIER_VL view.

Each hierarchy folder contains the hierarchy definition information and up to twenty levels of parent-child relationships. Within each level, the internal dimension member ID, member display code, and member name and description are available.

The FEM_DIS_NAT_ACCTS_HIER_VL view is a hierarchy transformation view based on the FEM_NAT_ACCTS_HIER table. LEVEL1 to LEVEL20 columns in FEM_DIS_NAT_ACCTS_HIER_VL view represent hierarchy levels 1 through 20.

- **EPF Company Cost Center Org Dimension Hierarchy:** This folder holds the details as defined in EPF for all of the hierarchies built for the Organizational Unit dimension.

Rows

The STD - FTP Interest Margin Account (Org/Product, Summary) report includes the following row:

- Natural Account Hierarchy

Related Topics

Overview of Oracle Transfer Pricing Reports, page 17-1

Oracle Transfer Pricing Reports Common Concepts, page 17-3

Generating and Viewing Oracle Transfer Pricing Reports , page 17-2

STD - FTP Margin Stratification

This report is based on the account level data and lets you define up to 10 transfer rate stratification ranges. The report shows both the matched spread and interest margin for each stratification range. The report comprises two Discoverer worksheets:

- **FTP Margin Stratification:** This worksheet provides a stratification on transfer rate for a single Calendar Period.
- **FTP Margin Stratification Over Time:** This worksheet provides the same information but with a stratification over time.

Business Area Folders

The STD - FTP Margin Stratification report is based on the following business area folder:

EPF All Account Tables: This is a custom folder that joins instrument tables and presents views that link them.

Joins

This Oracle Transfer Pricing report make use of the following additional joins to retrieve data:

- EPF Company Cost Center Organizations -> EPF All Account Tables
- EPF Natural Accounts -> EPF All Account Tables

Rows

Margin Stratification by Transfer Rate: DECODE(LEAST(GREATEST(Current Net Rate,:T1L),:T1H),Current Net Rate,'01. 0 -
'||:T1H,DECODE(LEAST(GREATEST(Current Net Rate,:T2L),:T2H),Current Net Rate,'02. '||:T2L||' - '||:T2H,DECODE(LEAST(GREATEST(Current Net Rate,:T3L),:T3H),Current Net Rate,'03. '||:T3L||' -
'||:T3H,DECODE(LEAST(GREATEST(Current Net Rate,:T4L),:T4H),Current Net Rate,'04. '||:T4L||' - '||:T4H,DECODE(LEAST(GREATEST(Current Net Rate,:T5L),:T5H),Current Net Rate,'05. '||:T5L||' -
'||:T5H,DECODE(LEAST(GREATEST(Current Net Rate,:T6L),:T6H),Current Net Rate,'06. '||:T6L||' - '||:T6H,DECODE(LEAST(GREATEST(Current Net Rate,:T7L),:T7H),Current Net Rate,'07. '||:T7L||' -
'||:T7H,DECODE(LEAST(GREATEST(Current Net Rate,:T8L),:T8H),Current Net Rate,'08. '||:T8L||' - '||:T8H,DECODE(LEAST(GREATEST(Current Net Rate,:T9L),:T9H),Current Net Rate,'09. '||:T9L||' -
'||:T9H,DECODE(LEAST(GREATEST(Current Net Rate,:T10L),:T10H),Current Net Rate,'10. '||:T10L||' - '||:T10H,'Other'))))))))

Totals

Grand Total Rows Sum for Record Count

Related Topics

Overview of Oracle Transfer Pricing Reports, page 17-1

Oracle Transfer Pricing Reports Common Concepts, page 17-3

Generating and Viewing Oracle Transfer Pricing Reports , page 17-2

A

Standard Navigation Paths

This appendix gives you information to navigate through the pages referred to in this guide.

This appendix covers the following topics:

- Standard Navigation Paths

Standard Navigation Paths

Although you may have customized your navigator, typical navigation paths are shown in this table. Access all of these pages through the Oracle Transfer Pricing Supervisor (FTP Supervisor) or Oracle Transfer Pricing User (FTP User) responsibility.

Page	Navigation Path
Transfer Pricing Rule Home	Rules > Calculation > Transfer Pricing
Transfer Pricing Rule Definition	Rules > Calculation > Transfer Pricing > Create Transfer Pricing Rules
Transfer Pricing Version Definition	Rules > Calculation > Transfer Pricing > Create Transfer Pricing Rules > Continue
Transfer Pricing Methodology	Rules > Calculation > Transfer Pricing > Create Transfer Pricing Rules > Continue > Update
Cash Flow Edits Definition	Rules > Calculation > Cash Flow Edits
Payment Pattern Home	Rules > Patterns > Payment Patterns
Repricing Pattern Home	Rules > Patterns > Repricing Patterns

Page	Navigation Path
Interest Rate Code Home	Rules > Interest Rate Codes
Monitor Requests	Process Management > Requests > Monitor
Rate Index Rule Home	Rules > Calculation > Rate Index
Prepayment Rule Home	Rules > Calculation > Prepayment
Prepayment Methodology	Rules > Calculation > Transfer Pricing > Create Transfer Pricing Rules > Continue > Update
Prepayment Table Rule Home	Rules > Calculation > Prepayment Table
Propagations Pattern Home	Rules > Patterns > Propagation
Transfer Pricing Process Home	Rules > Calculation > Transfer Pricing Process
Daily Rates Home	Rules > Currency Rates
Historical Rates Home	Rules > Currency Rates
Create Daily Rates	Rules > Currency Rates > Daily Rates
Create Historical Rates	Rules > Currency Rates > Historical Rates

Glossary

absolute payment pattern

A type of payment pattern commonly used for instruments that are on a seasonal schedule, such as agricultural or construction loans, and require special payment handling based on months or seasons.

See also: payment pattern, page Glossary-6

absolute repricing pattern

A type of repricing pattern commonly used for instruments with date dependent repricing characteristics.

See also: repricing pattern, page Glossary-8

Account table

Stores a detailed set of transaction-level data attributes pertaining to instruments. For example, origination date, outstanding balance, contracted rate, and maturity date.

An account table is also known as an instrument table.

assignment date

Indicates the relevant date for which the associated yield curve has to be referenced. This parameter used for defining the Spread from Interest Rate Code transfer pricing method. You can choose origination date, last reprice date, or the last day of the associated calendar period as the assignment date.

assumptions

A set of values or methodologies that apply to a single dimension value.

attributed dimension

A dimension whose members can have other properties or qualifiers known as *dimension attributes*. While attributed dimensions can also have hierarchies, they are not required to do so. For example, the Ledger dimension and Financial Element dimension do not have any hierarchies.

calculation mode

Any of the two ways, *standard* or *remaining term*, of calculating transfer rates and options costs supported by Oracle Transfer Pricing.

Cash Flow Edits

A business rule that allows you to verify and correct your Account table data.

cash flow edit logic

A set of checks that are performed in a specified order on the Account table data during a Cash Flow Edits rule run.

cash flow transfer pricing methods

Methods that generate transfer rates based on the cash flow characteristics of the instruments. These methods are typically used for instruments that amortize over time.

charge/credit accrual basis

The basis on which charge/credit accrues for a business unit and the offsetting treasury unit, similar to the accrual basis used in calculation of interest.

condition

A business object that filters the source data that is used as input to a business rule.

Data set

A dimension used for segregating data into different sets according to its use or its source, for example, to separate actuals data, budget data, and encumbrances data. Other uses include separating test data from production data and creating separate data sets for *what-if* analysis.

Data Inspector

A business rule that allows you to view or update data in a specific table or view registered in the Oracle Enterprise Performance Foundation data schema.

dimension

A structure that can be used to categorize business data. A dimension contains members. A dimension can be hierarchical in that you can organize the members into one or more hierarchies, or nonhierarchical.

dimension attribute

A property or qualifier that further describes a dimension member. An attribute can be anything such as a date, a number, or a character string. For example, the Geography dimension can have an attribute Population that designates how many people live in

that area. Each member of the Geography dimension therefore has an associated population.

dimension based rule

A business rule whose definition varies depending on the dimensional values of the data to which it is applied.

dimension identifier

A character string or a combination of character strings that uniquely identifies each member of a dimension. Dimension identifiers are nontranslatable, as they are the same regardless of the language context. Each dimension has its own unique set of columns in the Enterprise Performance Foundation interface tables that serve as the dimension identifier for that dimension.

dimension member

The values used to populate dimension columns in account, transaction, or statistical tables. Such values represent the individual organization units, distribution channels, products, and so on of which each dimension is comprised. In a hierarchy, both *lowest level* and *node level* values are considered to be dimension members.

driver

A variable that influences the prepayment behavior of an instrument. You can build a prepayment table using up to three prepayment drivers. Each driver maps to an attribute of the underlying transaction (age or term, or rate) so the cash flow engine can apply a different prepayment rate based on the specific characteristics of the record.

effective dated hierarchy

A hierarchy whose definition changes over time, also known as slowly changing hierarchies.

entered currency

The currency in which business transactions take place. Entered currency might be different from functional Currency.

See also: functional currency, page Glossary-3

fact table

A table that contains data uniquely differentiated by dimension columns.

FEM_BALANCES

See: Management Ledger table, page Glossary-5

functional currency

The currency in which an organization keeps its books of accounts. Functional currency is associated with a particular ledger.

hierarchy

A structure of dimension members organized by parent-child relationships.

hierarchy definition

A structure of dimension members organized by parent-child relationships, for a designated effective date range. Hierarchy definition is synonymous with hierarchy version, in that it is one instance of the hierarchy.

hierarchy object

A collection of hierarchy definitions. The individual hierarchy definitions represent a particular picture of the hierarchy object.

historical term

The period preceding the assignment date over which the average of daily interest rates from a yield curve is taken. This parameter is used in Moving Averages transfer pricing method.

instrument table

See: Account table, page Glossary-1

item

A name for data that is stored in your database. For example, the item *department* is the name for all the departments at your company. You select items in the Workbook Wizard to get the data you want. Discoverer uses these items to write a SQL query. When the database returns the data that answers the query, the items you chose appear as row and column headings in a spreadsheet-like format.

Interest Rate Codes

Allows you to define and manage historical interest rates and term structure parameters for various interest rate curves.

interpolation method

A prepayment rates lookup method. The interpolation method is used when the value of prepayment driver does not fall on the nodes defined for it. This method assumes that prepayment speeds change on a straight-line basis between the two nodes and calculates accordingly.

See also: lookup method, page Glossary-5

lag term

Indicates that interest rate should be referenced from a yield curve for a date earlier than the assignment date. Lag term is applicable to the Spread from Interest Rate Code transfer pricing method.

leaf

A node, in a hierarchy, that has no children. All dimension members are nodes, but not all nodes are lowest level dimension numbers.

ledger migration

The process for generating charges or credits, for funds provided or used, for migration to the Management Ledger table based on the transfer rates or option costs, obtained from transfer pricing calculation or propagation processing.

level

A property of hierarchical dimensions that designates a category of like members.

For example, in the Geography dimension there might be a level named City and a level named State. Geography members such as Tulsa and Dallas belong in the City level, while Geography members such as Texas and Oklahoma belongs in the State level. The designation of level is the same across all hierarchies within a dimension. In other words, Texas is always a state in all Geography hierarchies.

Line Item dimension

A Product dimension on which all Oracle Transfer Pricing product level assumptions are based. The Line Item dimension should be populated with your product chart of account at a level of detail appropriate for assigning transfer pricing assumptions to your data.

lookup method

Method used to calculate prepayment rates for a prepayment driver values that do not fall on the nodes defined for that particular prepayment driver.

low discrepancy sequences

Sequences, also known as quasirandom sequences, designed to fill the space uniformly. These achieve better accuracy with fewer scenarios than pseudorandom sequences when applied to numerical problems, integration in high dimension, and so on.

Management Ledger table

Also know as FEM_BALANCES, the most central fact table in the Oracle Enterprise Performance Foundation framework. It contains ledger and some statistical data and especially aggregated information, such as cash and other assets, and equity. This table supports Oracle Financial Services applications.

mid-period repricing option

Allows you to take in to account the impact of high market rate volatility while generating transfer prices for your products. However, the mid-period repricing option applies only to adjustable rate instruments and is available only for certain noncash flow transfer pricing methods.

node

A dimension value located anywhere in a hierarchy.

node level assumption

An assumption assigned to a dimension value at a level higher than a leaf level. A node level assumption is typically associated with a business rule that uses a hierarchical dimension.

See also: Line Item dimension, page Glossary-5

noncash flow transfer pricing methods

Transfer pricing methods that do not require the calculation of cash flows. While some of the noncash flow methods are available only with the Account tables data source, some are available with both the Account and Ledger table data sources.

option cost

The cost of optionality in terms of a spread over the transfer rate. Consider a mortgage that can be prepaid by the borrower at any time without penalty. Here the lender has granted the borrower an option to buy back the mortgage on par, even if interest rates have fallen in value. Thus, this option has a cost to the lender.

payment event

A set of payment characteristics, which define the time line and amount of a specific payment in a payment pattern.

payment pattern

A user-defined custom amortization pattern. Payment patterns allows the cash flow engine to correctly generate cash flows for instrument records that amortize in a nonstandard way. Payment Patterns are linked to instrument records through user-defined amortization type codes.

preparation

The phase in which the transfer pricing engine gathers information and prepares data structures for the run. This phase is only executed once per engine run.

prepayment methodologies

A set of methods used to model the prepayment behavior of amortizing instruments and quantify the associated prepayment risk.

prepayment risk

The possibility that borrowers might choose to repay part or all of their loan obligations before the scheduled due dates. Prepayments can be made by either accelerating principal payments, also known as curtailment, or refinancing.

Prepayment rule

A business rule used to manage the association of prepayment methodologies and rates to various product-currency combinations.

process control information

Information required to control processing, for example, a condition rule attached to a process rule. Process control information affects the rows or tables processed but not the results on those rows.

process data

Data required to produce results.

processing table

Account table available for, or included in, a Transfer Pricing Process rule run.

propagation pattern

Allows you to specify parameters, such as source tables, used in propagation of transfer rates and option costs, for any applicable instrument table from a prior period.

propagation process

The process for copying historical results, either transfer rates or option costs, or both, that were generated by the application in a previous run for a prior period, to the current period records.

rate conversion

A process, involving the use of conversion formula, for transforming interest rates from their starting format into a format proper for their use in any given process.

Rate Index rule

A business rule used to establish a relationship between an Interest Rate Code, typically a risk-free yield curve, and other Interest Rate Codes. Rate Index rule is used to generate forward rates for option cost calculations using stochastic interest rate models.

random number generation method

Method to determine how the Monte Carlo process selects random numbers. The random number generation method has two variations, low discrepancy and pseudorandom sequences.

range method

A prepayment rates lookup method. Under this method, the prepayment rates are determined by calculating a range of values on an axis. This method assumes that the prepayment speed remains the same for the entire range.

rate lookup

The procedure for deriving a transfer rate for the appropriate date-term combination from a particular yield curve.

rate spread

The fixed positive or negative spread from an Interest Rate Code or Note Rate, used to generate transfer rates in the Spread from Interest Rate and Spread from Note Rate methods.

reference currency

Currency in which the instrument data is expressed and designated by the currency code on the record. Within the application, the reference currency must be selected to indicate assumptions that will be applied to corresponding currency designations contained in the account data.

relative payment pattern

A type of payment pattern commonly used for modeling instruments with irregular payment frequencies or for instruments where the payment type changes over time.

See also: payment pattern, page Glossary-6

relative repricing pattern

A type of repricing pattern comprising a series of repricing events driven by user defined time lines. A relative repricing pattern is used for instruments where the repricing is determined by elapsed time since origination.

See also: repricing pattern, page Glossary-8

remaining term calculation mode

Allows you to calculate transfer rates and option costs for instrument records based on the remaining term of the instrument from the as of date of the data, rather than the origination date or last repricing date of the instruments. This mode is one of the two calculation modes supported by Oracle Transfer Pricing.

repricing pattern

A user-defined custom repricing pattern. Repricing patterns allow the cash flow engine to correctly generate interest for instrument records that reprice in a nonstandard way. Repricing patterns are linked to instrument records through user-defined adjustable type codes.

rule

A grouping of dimension values with their assumptions, also known as a business rule.

Rule of 78

An approach used by banks to formulate a loan amortization schedule. Also known as The Rule of the Sum of the Digits, this method of computing unearned interest is used on installment loans with add-on interest. The number 78 is based on the sum of the digits from 1 to 12. This approach causes a borrower to pay more interest at the beginning of the loan when there is more money owed and less interest as the obligation is reduced.

simple dimension

A dimension that does not have hierarchies or attributes. A simple dimension is just a list of members.

smoothing method

Method used to interpolate rates on the valuation curve for terms that fall between given points.

split payment pattern

A type of payment pattern containing both absolute and relative payment events under a single amortization code.

See also: payment pattern, page Glossary-6

spread

The difference between the customer rate and the transfer rate or market rate (determined by a reference IRC).

standard calculation mode

Allows you to calculate transfer rates for instrument records based on the origination or last repricing date of the instruments. You can also use it to calculate option costs based on the origination date. This mode is one of the two calculation modes supported by Oracle Transfer Pricing.

terminal condition

A condition that has no other conditions nested under it.

Term Structure Model

Model for governing the generation of forward stochastic rates, discount factors for each scenario, and discrete rates for any maturity used in calculating the option-adjusted spread.

transfer pricing methodologies

A set of methods used to generate transfer rates for different types of instruments, including amortizing and nonamortizing instruments.

Transfer Pricing rule

A business rule used to manage the association of transfer pricing methodologies and certain parameters used in option costing to various product-currency combinations.

Transfer Pricing Process rule

A business rule used to formulate and execute transfer pricing or option cost processing requests.

user-defined dimension

A dimension that enables additional customization, beyond the standard dimensions provided by Oracle Enterprise Performance Foundation. Enterprise Performance Foundation provides user-defined dimensions that support hierarchies and attributes, as well as user-defined simple dimensions.

user-defined payment pattern

See: payment pattern, page Glossary-6

user-defined repricing pattern

See: repricing pattern, page Glossary-8

value set

A set of dimension members. Value sets are a component of the dimension identifier for some dimensions. The value set concept enables larger organizations to employ the same code to identify multiple dimensions across different regions. For example, in Europe, code 123 identifies Customer A, while the same code in the United States identifies Customer B.

version

A version of any of the Oracle Transfer Pricing business rules. Each rule version is

effective dated and can also be named.

workbooks

A collection of worksheets in Oracle Discoverer. A workbook contains data that is related in some way but organized to show different perspectives.

For example, you might decide to create a workbook to show the sales history for Product A. However one worksheet could show sales for last month, another worksheet could show sales compared to the same month five years ago, and another could show sales per region. All three worksheets contain sales data related to Product A, but each is organized to show a different perspective.

worksheet

A page or tab in Oracle Discoverer. A worksheet contains rows and columns of data and allows you to analyze and share it. Each worksheet is created by its own query. Every time you open or refresh a worksheet, Oracle Discoverer sends its query to the database to get the most current data.

yield curve term

The point on the yield curve that the system references to calculate transfer rates.

Index

A

- absolute payment patterns
 - usage, 2-47
- absolute repricing patterns
 - usage, 2-51
- Account tables, 2-1
- analyzing transfer pricing results
 - general steps, 2-57
 - generating detailed queries, 2-57
 - reviewing and comparing the funding center impact, 2-57
 - reviewing historical rates, 2-57
 - using the detailed cash flow audit option, 2-57
- audit options
 - 1 Month Rates, 2-39
 - Detailed Cash Flows, 2-39
 - Forward Rates, 2-39
- average balance processing
 - defining daily rates, 5-6, 5-7
 - entering historical rates, 5-23
 - notes on translating average balances, 5-53
 - revaluing balances, 5-27
 - translating balances, 5-36
- average balances
 - notes on translating, 5-53
 - revaluing, 5-27
 - translating, 5-36

C

- calculation modes
 - Remaining Term, 2-42

- Standard, 2-42
- caps, 2-37
- Cash Flow Edits
 - creating cash flow edit rules, 6-2
 - definition, 2-52
 - deleting cash flow edit rules, 6-1
 - duplicating cash flow edit rules, 6-1
 - executing cash flow edit rules, 6-4
 - searching for cash flow edit rules, 6-1
 - selecting preview, 2-52
 - viewing and updating cash flow edit rules, 6-1
 - viewing results, 2-52
- conditional assumptions
 - availability, 2-22
 - inserting methods
 - guidelines, 11-10
 - prerequisites, 11-9
 - procedure, 11-9
 - structure
 - ELSE block, 2-25
 - ELSE IF block, 2-24
 - IF block, 2-24
 - usage, 2-21
- conversion rates, 5-11
- conversion rate types
 - corporate, 5-12
 - defining, 5-11
 - spot, 5-12
 - user, 5-12
- Conversion Rate Types window, 5-11
- corporate conversion rate type, 5-12
- creating cash flow edit rules
 - procedure, 6-2

- selected terminology, 6-2
- creating conditional assumptions
 - guidelines, 11-8
 - prerequisites, 11-3
 - procedure, 11-3
 - defining additional clauses, 11-7
 - inserting logic into blocks, 11-11
 - updating clauses, 11-12
 - validating and saving a conditional assumption, 11-8
 - process overview, 11-1
 - selected terminology, 11-3
- creating Interest Rate Codes
 - guidelines, 9-3
 - procedure, 9-3
- creating payment patterns
 - procedure, 7-3
- creating Prepayment Table rules
 - assigning node values, 14-5
 - creating rule and version, 14-2
 - defining prepayment table structure, 14-4
- creating repricing patterns
 - procedure, 8-3
- cross rates, generating, 5-57
- currencies, 5-9
 - Currency Rates Manager, 5-57
 - defining, 5-9
 - enabling or disabling, 5-11
 - ISO predefined in GL, 5-9
 - revaluing balances, 5-27
 - translating balances, 5-36
 - working with multiple, 5-8
- Currencies window, 5-9, 5-11
- currency conversion
 - defining rate types, 5-11
 - entering daily rates, 5-13
 - entering historical rates, 5-23
- Currency Rates Manager, 5-57
 - cross rates, 5-57
 - daily rates spreadsheet interface, 5-61
 - historical rates spreadsheet interface, 5-62

D

- daily conversion rates
 - entering, 5-13
 - loading automatically, 5-18

- daily rates spreadsheet interface, 5-61
- Daily Rates window, 5-13
- data reconciliation
 - comparing ending balances, 2-59
 - correcting variances, 2-59
 - defining levels, 2-59
 - process, 2-59
- defining absolute payment patterns
 - guidelines, 7-6
 - prerequisites, 7-4
 - procedure, 7-4
 - selected terminology, 7-4
- defining absolute repricing patterns
 - prerequisites, 8-4
 - procedure
 - flat repricing, 8-5
 - indexed repricing, 8-6
 - selected terminology, 8-4
- defining prepayment methodologies
 - arctangent calculation method, 13-9
 - constant prepayment method, 13-6
 - inputting parameters
 - all currencies for one product, 13-3
 - all products for one currency, 13-3
 - prepayment table method, 13-8
 - prerequisites, 13-3
 - procedure, 13-3
 - selected terminology, 13-3
- defining rate indexes
 - procedure, 10-2
 - adding index term definitions, 10-5
 - adding the index, 10-5
 - selected terminology, 10-2
- defining relative payment patterns
 - guidelines, 7-9
 - prerequisites, 7-7, 7-10
 - procedure, 7-7
 - selected terminology, 7-7
- defining relative repricing patterns
 - prerequisites, 8-7
 - procedure, 8-7
 - selected terminology, 8-7
- defining the arctangent calculation method
 - prerequisites, 13-8, 13-9
 - procedure, 13-8, 13-9
- defining the constant prepayment method
 - prerequisites, 13-6

- procedure, 13-6
- defining the redemption curve methodology
 - prerequisites, 12-9
 - procedure, 12-9
- defining the unpriced account methodology
 - prerequisites, 12-10
 - procedure, 12-10
- defining transfer pricing methodologies
 - guidelines, 12-7
 - inputting parameters
 - all currencies for one product, 12-3
 - all products for one currency, 12-3
 - prerequisites, 12-4
 - procedure, 12-4
 - redemption curve methodology, 12-9
 - selected terminology, 12-4
 - unpriced account methodology, 12-10
- Detailed Cash Flows audit options
 - determining the value of Object ID, 2-39
 - FTP_PROCESS_CASH_FLOWS table, 2-39
 - viewing data, 2-39

E

- EPF
 - See* Oracle Enterprise Performance Foundation
- exchange rates
 - corporate, 5-12
 - spot, 5-12
 - user, 5-12
- executing cash flow edit rules
 - prerequisites, 6-4
 - procedure, 6-4
 - procedure to determine the Object ID, 6-4
- executing Transfer Pricing Process rules
 - prerequisites, 16-9
 - procedure, 16-9
 - procedure to determine the Object ID, 16-14
 - selected terminology, 16-9
- extracting
 - rules, 3-9

F

- FEM_BALANCES table
 - See* Management Ledger table
- floors, 2-37

- foreign currency
 - daily rates, 5-13
 - notes on translating average balances, 5-53
 - revaluing balances, 5-27
 - translation, 5-36

FTP

- See* Oracle Transfer Pricing

- FTP_PROCESS_CASH_FLOWS audit table, 2-55
- future interest rates, 2-38

G

- generating cross rates, 5-57

H

- historical rates
 - automatically assigned rate types, 5-26
 - entering, 5-23
- historical rates spreadsheet interface, 5-62
- Historical Rates window, 5-23

I

- indexes, 2-38
 - See also* Interest Rates Codes
- index terms, 10-2
- Instrument tables
 - See* Account tables
- Interest Rate Codes
 - creating, 9-3
 - definition, 2-53
 - deleting, 9-1
 - loading data
 - using UI, 9-6
 - using Web ADI, 9-8
 - rate attributes, 2-53
 - rate lookups, 2-53
 - searching, 9-2
 - structure modeling parameters, 2-53
 - viewing and updating, 9-1

L

- ledger migration options
 - option costs, 2-43
 - transfer rates, 2-43
- ledger migrations
 - purpose, 2-43

- loading
 - rules, 3-9
- loading data using UI
 - guidelines, 9-8
 - loading historical rates, 9-6
 - prerequisites, 9-6
 - procedure
 - loading historical rates, 9-6
 - loading parameters, 9-7
- loading data using Web ADI
 - prerequisites, 9-8
 - procedure, 9-8

M

- Management Ledger table, 2-1
- mid-period repricing options
 - availability, 2-14
 - computations, 2-14
 - exceptions to typical calculations
 - origination date exception, 2-18
 - teased loan exception, 2-17
 - typical calculations, 2-15
 - usage, 2-14
- multi-currency
 - overview, 5-3
- multi-currency accounting
 - overview, 5-3
- multiple currencies
 - working with, 5-8

N

- node level assumptions
 - behavior, 2-19
 - usage, 2-19

O

- OFS
 - See* Oracle Financial Services applications
- options costs
 - definition, 2-37
 - Monte Carlo technique, 2-37
- Oracle Approvals Management (AME), 4-1
- Oracle Enterprise Performance Foundation, 2-1
- Oracle Financial Services applications
 - Oracle Budgeting and Planning, 1-2

- Oracle Enterprise Performance Foundation, 1-2
- Oracle Profitability Manager, 1-3
- Oracle Risk Manager, 1-2
- Oracle Transfer Pricing, 1-1
- Oracle Transfer Pricing
 - integration with Oracle Financial Services applications, 1-3
 - key benefits, 1-1
 - overview, 1-1
- Oracle Transfer Pricing Process
 - overview of steps, 2-1
 - accessing transfer pricing detail cash flow results for audit purposes, 2-55
 - analyzing results, 2-57
 - capturing instrument behavior, 2-44
 - See also* payment patterns
 - deciding on and managing historical rates information, 2-53
 - See also* Interest Rate Codes
 - performing Cash Flow Edits, 2-52
 - See also* Cash Flow Edits
 - reconciling the data, 2-59
 - reprocessing erroneous accounts, 2-59
 - reviewing processing errors, 2-58
 - setting and executing the Transfer Pricing Process rule, 2-36
 - See also* Transfer Pricing Process rules
 - setting Prepayment rules, 2-27
 - See also* Prepayment rules
 - setting Transfer Pricing rules, 2-3
 - See also* Transfer Pricing rules
- Oracle Transfer Pricing reports
 - common concepts, 17-3
 - business area folders, 17-3
 - joins, 17-3
 - parameters, 17-4
 - report headings and calculations, 17-4
 - rows, 17-4
 - discoverer file names, 17-1
 - generating and viewing, 17-2
 - procedure, 17-2
 - list of reports, 17-1
 - Overview, 17-1
 - reporting solution , 17-1
 - STD - FTP Interest Margin Account (Org/Product, Summary) report

- business area folders, 17-6
- overview, 17-6
- rows, 17-6
- STD - FTP Interest Margin Account report
 - business area folders, 17-5
 - overview, 17-5
 - rows, 17-5
- STD - FTP Margin Stratification report
 - business area folders, 17-7
 - joins, 17-7
 - overview, 17-6
 - rows, 17-7
 - totals, 17-8
- Oracle Workflow, 4-1
- owners' equity accounts
 - notes on translating, 5-50

P

- payment characteristics
 - payment methods, 2-45
 - value, 2-45
- payment methods
 - absolute payment, 2-45
 - interest only, 2-45
 - percentage of current balance, 2-45
 - percentage of current payment, 2-45
 - percentage of original balance, 2-45
 - percentage of original payment, 2-45
- payment patterns
 - amortization code, 2-44
 - creating, 7-3
 - deleting, 7-1
 - duplicating, 7-1
 - payment events, 2-45
 - See also* payment characteristics
 - payment pattern code, 2-44
 - relevance, 2-44
 - searching, 7-2
 - structure
 - absolute, 2-44
 - relative, 2-44
 - split, 2-44
 - viewing and updating, 7-1
- payment values
 - interest payments, 2-47
 - principal payments, 2-47

- prepayment drivers
 - age/term drivers, 2-29
 - interest rate drivers, 2-29
- prepayment methods
 - arctangent calculation method, 2-31
 - constant prepayment method, 2-28
 - prepayment table method, 2-28
- prepayment risks
 - definition, 2-27
 - impact, 2-27
- Prepayment rules
 - compatible transfer pricing methods, 2-27
 - conditional assumptions, 2-35
 - See also* creating conditional assumptions
 - creating
 - defining prepayment methodologies, 13-3
 - procedure, 13-2
 - deleting, 13-1
 - duplicating, 13-1
 - methodologies, 2-27
 - See also* defining prepayment methodologies
 - node level assumptions, 2-35
 - searching, 13-1
 - usage, 13-1
 - viewing and updating, 13-1
- Prepayment Table rules
 - creating
 - procedure, 14-2
 - selected terminology, 14-2
 - deleting, 14-1
 - duplicating, 14-1
 - searching, 14-1
 - usage, 14-1
 - viewing and updating, 14-1
 - See also* updating prepayment tables
- prepayment tables
 - diagram, 2-30
 - structure
 - interpolation or range methods, 2-28
 - prepayment drivers, 2-28
 - prepayment nodes, 2-28
- processing errors
 - error log, 2-58
 - rectification, 2-58

- programs
 - revaluation, 5-27
- propagation
 - propagation pattern , 15-1
 - relevance, 2-39
- propagation options
 - option costs, 2-39
 - transfer rate, 2-39
- propagation pattern
 - defining the propagation pattern
 - procedure, 15-1
 - selected terminology, 15-1
 - propagating transfer pricing results
 - prerequisites, 15-3
 - procedure, 15-3

R

- Rate Index rules
 - creating, 10-1
 - defining rate indexes, 10-2
 - deleting, 10-1
 - duplicating, 10-1
 - searching, 10-1
 - usage, 2-38
 - viewing and updating, 10-1
- rate lookups
 - date used, 2-53
 - endpoints, 2-53
 - example, 2-53
 - relevance, 2-53
 - term used
 - multiplier, 2-53
 - yield curve, 2-53
- rate risk spreads
 - current rate risk spread, 2-42
 - embedded rate risk spread, 2-42
 - total rate risk spread, 2-42
- rates, entering, 5-57
- relative payment patterns, 2-48
 - usage, 2-48
- relative repricing patterns, 2-52
 - usage, 2-52
- reports
 - See* Oracle Transfer Pricing reports
- repricing events
 - repricing types

- flat rate, 2-50
- indexed rate, 2-50
- repricing patterns
 - components, 2-48
 - creating, 8-3
 - deleting, 8-1
 - duplicating, 8-1
 - elements affecting repricing, 2-48
 - relevance, 2-48
 - repricing events, 2-49
 - searching, 8-2
 - types
 - absolute repricing patterns, 2-49
 - relative repricing patterns, 2-49
 - viewing and updating, 8-1
- reprocessing instrument data
 - line item dimensions members, 2-59
 - unpriced accounts, 2-59
- revaluation
 - program, 5-27
 - summary of Revaluation Program, 5-33
 - tracking by balancing segment and cost center, 5-27
- Revalue Balances window, 5-27
- rules
 - approval process for, 4-2
 - approval process for, as related to production data sets, 4-1
 - create, 3-6
 - deleting, 3-8
 - deletion process for, 4-2
 - duplicating, 3-7
 - extracting, 3-9
 - home page, 3-2
 - loading, 3-9
 - search, 3-5
 - updating, 3-7
 - viewing, 3-7

S

- searching for Interest Rate Codes
 - prerequisites, 9-2
 - procedure, 9-2
- searching for payment patterns
 - prerequisites, 7-2
 - procedure, 7-2

- searching for repricing patterns
 - prerequisites, 8-2
 - procedure, 8-2
- setting up
 - multi-currency accounting, 5-3
- SFAS 52, 5-23, 5-27
- split payment patterns, 2-48
 - usage, 2-48
- spot conversion rate type, 5-12
- spreads
 - option adjusted spread, 2-37
 - static spread, 2-37
- standard navigation paths, A-1

T

- transfer pricing methodologies
 - cash flow methods, 2-4
 - duration, 2-5
 - weighted average cash flow, 2-6
 - zero coupon pricing, 2-7
 - mid-period repricing options, 2-14
 - noncash flow methods, 2-4
 - moving averages, 2-9
 - redemption curve, 2-12
 - spread from interest rate code, 2-11
 - spread from note rate, 2-11
 - straight term, 2-10
 - unpriced account, 2-13
- Transfer Pricing Process rules
 - Account table fields, 2-36
 - creating
 - prerequisites, 16-2
 - procedure, 16-2
 - selected terminology, 16-2
 - deleting, 16-1
 - duplicating, 16-1
 - executing, 16-9
 - searching, 16-1
 - selecting audit options, 2-39
 - selecting migration options, 2-43
 - selecting propagation options, 2-39
 - specifying option cost parameters, 2-37
 - usage, 16-1
 - using different calculation modes, 2-42
 - using Rate Index rules, 2-38
 - viewing and updating, 16-1

- Transfer Pricing reports
 - See* Oracle Transfer Pricing reports
- Transfer Pricing rules
 - creating
 - defining transfer pricing methodologies, 12-3
 - procedure, 12-2
 - deleting, 12-1
 - duplicating, 12-1
 - methodologies, 2-4
 - See also* defining transfer pricing methodologies
 - searching, 12-1
 - usage, 12-1
 - using conditional assumptions, 2-21
 - See also* conditional assumptions
 - using node level assumptions, 2-19
 - See also* node level assumptions
 - viewing and updating, 12-1
- Translate Balances window, 5-36
 - notes on translating average balances, 5-53
- translation
 - cumulative translation adjustment account, 5-40
 - foreign currency, 5-36
 - notes on translating average balances, 5-53
 - notes on translating owners' equity accounts, 5-50
 - notes on translating revenue/expense accounts, 5-51
 - period-to-date vs. year-to-date translation rules, 5-37
 - rates used for remeasurement, 5-38
 - rates used for translation, 5-38
 - restating previously translated balances, 5-52
 - translating retained earning account, 5-50
 - translation vs. remeasurement, 5-39
 - with historical rates and amounts, 5-49

U

- updating prepayment tables
 - prerequisites, 14-6
 - procedure, 14-6
 - adding prepayment rates, 14-7
 - updating dimensions values, 14-
 - updating speeds, 14-6

See also adding prepayment rates
user conversion rate type, 5-12
user-defined payment patterns, 7-1
See also payment patterns
user-defined repricing patterns, 8-1
See also repricing patterns

V

valuation curves, 10-2
viewing cash flows audit data
 creating Conditions, 2-39
 creating Data Inspector rules, 2-39