Oracle® Financial Services Analytical Applications

Funds Transfer Pricing User Guide



Release 8.1.2.11.0
G59127-01
August 2026



Oracle Financial Services Analytical Applications Funds Transfer Pricing User Guide, Release 8.1.2.11.0

G59127-01

Copyright © 2011, 2026, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Contents

1 Oracle Financial Services Funds Transfer Pricing

2 Overview of OFSAA Infrastructure

2.1 Overview of Dimensionality in OFSAA 2
2.1.1 Referential Integrity 2
2.1.2 Dimension Attributes 3
2.1.3 Hierarchies 3
2.1.4 Rollup Members for Dimensions Supporting Hierarchies 4
2.1.5 Dimension Types 4
2.1.5.1 Key Processing Dimensions 4
2.1.5.2 Standard Dimensions 5
2.1.5.3 Simple Dimensions 5
2.2 Attributes 7
2.2.1 Unbounded and Bounded Attribute Data Types 7
2.3 Overview of OFSAA Rules 7
2.3.1 Dependencies 8
2.4 OFSAA Filters 8
2.5 OFSAA Expressions 9
2.6 Folders 9
2.6.1 Visibility of Rules 10
2.6.2 Access Type 10
2.7 Global Preferences 10
2.8 System Configuration 11
2.9 Application Preferences 11

3 OFSAA Attributes

3.1 Adding Attribute Definition 1
3.2 Attribute Definition Operations 3

4 Dimension Members

4.1 Adding Member Definition 2

Copyright © 2011, 2026, Oracle and/or its affiliates.

4.2 Viewing Member Definition 4
4.3 Modifying Member Definition 5
4.4 Copying Member Definition 5
4.5 Member Definition Dependencies 5
4.6 Deleting Member Definition 6

5 Hierarchy Maintenance

5.1 Adding Hierarchy Definition 2
5.2 Viewing Hierarchy Definition 6
5.3 Modifying Hierarchy Definition 6
5.4 Copying Hierarchy Definition 7
5.5 Hierarchy Definition Dependencies 7
5.6 Deleting Hierarchy Definition 7

6 Filter

6.1 Adding Filter Definition 2
6.1.1 Define Data Element Filter 5
6.1.2 Define Hierarchy Filter 9
6.1.3 Define Group Filter 11
6.1.4 Define Attribute Filter 11
6.2 Viewing Filter Definition 12
6.3 Modifying Filter Definition 12
6.4 Copying Filter Definition 12
6.5 Checking Dependencies 12
6.6 Viewing SQL of Filter 13
6.7 Deleting Filter Definition 13

7 Expression

7.1 Adding Expression Definition 2
7.2 Viewing Expression 4
7.3 Modifying Expression 4
7.4 Copying Expression 4
7.5 Checking Dependencies 5
7.6 Deleting Expression 5

8 Simplified Batches

8.1 Standard OFSAA Infrastructure Batching Functionality 1
8.1.1 Batch Maintenance 2

Copyright © 2011, 2026, Oracle and/or its affiliates.

8.1.2 Simplified Batches 3
8.1.2.1 Searching for a Simplified Batch 3
8.2 Simplified Batch Details 5
8.3 Creating a Simplified Batch 5
8.3.1 Batch Execution Type 6
8.3.2 Searching Task Name 7
8.3.3 Task Details 7
8.3.4 Task Selection 8
8.4 Running a Simplified Batch 10
8.5 Viewing the Task Logs 11
8.6 Additional Batch Options 11

9 OFSAA Rate Management

9.1 Interest Rates 2
9.1.1 Searching for Interest Rate Curve 2
9.2 Interest Rates Details 4
9.2.1 Creating an Interest Rate Code 4
9.2.2 Interest Rate Code Tabs 6
9.2.2.1 Navigating Between Interest Rate Code Tabs 6
9.2.2.2 Parameters Tab 14
9.2.2.3 Hybrid Term Structure Tab 17
9.2.3 Batch Creation 20
9.2.3.1 Running Hybrid Rate Generation using Simplified Batch 21
9.2.3.2 Viewing Log Messages 22
9.3 Currency 22
9.3.1 Searching for Currency 22
9.3.2 Editing Currencies 24
9.3.3 Adding Currencies 24
9.3.4 Deleting Currencies 24
9.4 Currency Rates 24
9.4.1 Preparing to Work with Exchange Rate Data 25
9.4.2 Adding Exchange Rate Data 26
9.4.3 Adding Multiple Exchange Rates 27
9.4.4 Editing Exchange Rate Data 27
9.4.5 Viewing Exchange Rate Data 27
9.4.6 Deleting Exchange Rate Data 27
9.4.7 Data Loader 27
9.4.8 Excel Compatibility 27
9.5 Currency Exchange Rate Validation 28
9.5.1 Features of Exchange Rate Validation 29
9.5.2 Currency Rate Validation 29

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.5.3 Validating Exchange Rate Relationships 30
9.5.4 Running an Exchange Rate Validation 31
9.5.4.1 Running a Validation Immediately 31
9.5.5 Scheduling One or More Validations 33
9.5.6 Viewing the Messages 34
9.6 Economic Indicators Summary Page 34
9.6.1 Searching for Economic Indicator 35
9.6.2 Economic Indicators Details 36
9.6.3 Adding an Economic Indicator 36

10 Holiday Calendar

10.1 Searching for a Holiday Calendar 1
10.2 Creating a Holiday Calendar 2
10.3 Excel Import and Export 4
10.4 Executing Holiday Calendar 4
10.5 Holiday Exceptions 4

11 OFS Funds Transfer Pricing Overview

11.1 Define Transfer Pricing Rules 4
11.2 Transfer Pricing Methodologies and Rules 4
11.2.1 Cash Flow: Average Life 7
11.2.2 Cash Flow: Duration 9
11.2.3 Cash Flow: Weighted Term 10
11.2.4 Cash Flow: Zero Discount Factors 11
11.2.5 Moving Averages 13
11.2.6 Straight Term 15
11.2.7 Spread from Interest Rate Code 16
11.2.8 Spread from Note Rate 17
11.2.9 Redemption Curve 18
11.2.10 Reference Period's Compounded Rate 18
11.2.11 Tractor Method 19
11.2.12 Caterpillar 22
11.2.13 Weighted Average Perpetual 23
11.2.14 Unpriced Account 24
11.2.15 Risk Free Reference Rates 25
11.2.16 Transfer Pricing Methods and the Mid-Period Repricing Option 27
11.2.16.1 Mid-Period Repricing Computations 28
11.2.16.2 Typical Calculations 28
11.2.16.3 Exceptions to Typical Calculations 30
11.2.17 Defining Transfer Pricing Methodologies Using Node Level Assumptions 31

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.2.18 Associating Conditional Assumptions with Assumption Rules 33
11.3 Define Prepayment Rules 36
11.3.1 Prepayment Methodologies and Rules 37
11.3.1.1 Constant Prepayment Method 38
11.3.1.2 Prepayment Model Method 38
11.3.1.3 PSA Method 40
11.3.1.4 Arctangent Calculation Method 40
11.3.2 Associating Node Level and Conditional Assumptions with Prepayment Rules 44
11.4 Define Adjustment Rules 44
11.4.1 Adjustment Methods and Rules 45
11.4.1.1 Adjustment - Fixed Rate 46
11.4.1.2 Adjustment - Fixed Amount 47
11.4.1.3 Adjustment - Formula Based Rate 48
11.4.1.4 Adjustment - Use TP Method from Selected TP Rule 50
11.4.2 Breakage Charges 53
11.4.2.1 Breakage Charge – Economic Loss 54
11.4.2.2 Breakage Charge – Fixed Amount 56
11.4.2.3 Breakage Charge - Fixed Percentage 56
11.4.2.4 Prerequisites for calculating Breakage Charges 56
11.4.2.5 Associating Node Level and Conditional Assumptions with Adjustment Rules 57
11.5 Define Alternate Rate Output Mapping Rules 57
11.6 Define and Execute a Transfer Pricing Process 57
11.6.1 Transfer Pricing Process and the Propagation Pattern 60
11.6.2 Transfer Pricing Process and Audit Options 62
11.6.3 SQL Query Results 62
11.6.4 Data Verification 63
11.6.5 Transfer Pricing Process Rule and Calculation Modes 63
11.6.6 Transfer Pricing Process and Migration Options 64
11.6.7 Transfer Pricing Process and Calculating Economic Value 66
11.6.8 Stochastic Transfer Pricing Process and Option Cost Parameters 67
11.6.9 Stochastic Transfer Pricing Process and Stochastic Rate Indexing Rules 68
11.7 Define and Execute a Break Identification Process 68
11.7.1 Full Breaks 70
11.7.2 Partial Breaks 71
11.7.3 Change in Attributes Break 74
11.8 Ledger Migration Process 75
11.8.1 Overview of the Ledger Migration Process 77
11.8.2 Understand the Ledger Migration 78
11.8.2.1 Transfer Rate, Adjustment Rate, and Option Cost Calculation 79
11.8.2.2 Charge/Credit Generation 80
11.8.2.3 Direct Transfer Pricing of Ledger Balances 80

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.8.2.4 Ledger Migration and the Management Ledger Table 81
11.8.2.5 Data Signage 81
11.8.2.6 Management Ledger Table Editing Standards 82
11.8.2.7 Ledger Migration and the Virtual Memory Table 82
11.8.3 Requirements for Successful Ledger Migration 82
11.8.3.1 Application Preferences 84
11.8.3.2 Dimensions 84
11.8.3.3 Entered and Local Currency 84
11.8.3.4 Source of Currency and Exchange Rate Information 84
11.8.3.5 Transfer Pricing Rule 85
11.8.3.6 Adjustment Rule 85
11.8.3.7 Product/Source Selection 85
11.8.3.8 Ledger Migration and Product Dimension Set Up 86
11.8.3.9 Offset Org Unit 86
11.8.3.10 Transfer Pricing Process Rule 86
11.8.3.11 Calculation Mode 86
11.8.4 Instrument Tables Accumulation 89
11.8.5 Management Ledger Table Processing 89
11.8.6 Transfer Pricing Accounts with Ledger-Only Data Source 90
11.8.7 Transfer Pricing Un-priced Accounts 90
11.8.8 Ledger Migration of Option Costs 91
11.8.9 Usage of Intermediate Tables in the Engine for Management Ledger 92
11.8.10 Migration Options- Functional Currency and Entered and Functional Currency 92
11.8.11 FSI_DB_INFO Dependency for Multi-Currency 93
11.9 Define Behavior Patterns 93
11.9.1 Behavior Pattern Structure 94
11.9.2 Payment Events 95
11.10 Define Payment Patterns 96
11.10.1 Payment Pattern Structure 96
11.10.2 Payment Events 97
11.10.3 Value 98
11.10.4 Absolute Payment Patterns 98
11.10.5 Relative Payment Patterns 99
11.10.6 Split Payment Patterns 99
11.11 Define Repricing Patterns 99
11.11.1 User-Defined Repricing Pattern 100
11.11.2 User-Defined Repricing Event 100
11.11.3 Absolute Repricing Pattern 102
11.11.4 Relative Repricing Pattern 102
11.11.5 Pricing with Balance Tiers 102
11.12 Perform Cash Flow Edits 103
11.13 Create Interest Rate Codes 103

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.13.1 Interest Rate Codes and Rate Lookups 104
11.14 Access Transfer Pricing Detail Cash Flow Results for Audit Purposes 105
11.15 Access Transfer Pricing Interest Rate Audit Results 106
11.16 Analyze Results 106
11.17 Review Processing Errors 107
11.18 Reprocess Erroneous Accounts 107
11.19 Reconcile the Data 107

12 Transfer Pricing Concept

12.1 Traditional Approaches to Transfer Pricing 2
12.2 A New Approach to Transfer Pricing 3
12.2.1 Advantages of Matched Rate Transfer Pricing 4
12.3 How Matched Rate Transfer Pricing Works? 5
12.4 Evaluating Interest Rate Risk 5

13 Common Rule Management Tasks

13.1 The Rule Summary Page 2
13.2 Search for Rules 3
13.3 Create Rules 4
13.4 View and Edit Rules 5
13.5 Copy Rules 6
13.6 Delete Rules 6
13.7 Dependency Checking 6

14 Application Preferences

15 Funds Transfer Pricing Configuration

15.1 Setup Information 1
15.2 Parallel Execution Configuration 2
15.3 Setup Parameters 3
15.4 Management Ledger Configuration 3

16 Cash Flow Edits

16.1 Creating Cash Flow Edit Processes 2
16.2 Executing Cash Flow Edit Processes 4

Copyright © 2011, 2026, Oracle and/or its affiliates.

17 Cash Flow Edits New

17.1 Overview of Cash Flow Edit Processes 1
17.2 Creating Cash Flow Edit Process 3
17.3 Executing Cash Flow Edit Process 5
17.3.1 Using Cash Flow UI 6
17.3.2 Batch Based Procedure 6
17.3.2.1 Simplified Batch 6
17.3.2.2 ICC Batch 7
17.3.2.3 Run Rule Framework 8
17.3.3 Using the Command Line 8
17.3.4 View Cash Flow Edit Process Log 10
17.3.5 Cash Flow Edit Process Errors 10

18 User Defined Behavior Patterns

18.1 Creating Behavior Patterns 1
18.1.1 Searching for Behavior Patterns 3
18.1.2 Defining Non-Maturity Behavior Patterns (for Tractor TP Method use) 4
18.1.3 Defining Non-Performing Behavior Patterns 8
18.1.4 Defining Devolvement and Recovery Behavior Patterns 10
18.1.5 Defining Non-Maturity Behavior Patterns 11
18.2 Export and Import Replicating Portfolio Data in Excel 14

19 User Defined Payment Patterns

19.1 Search for Payment Patterns 1
19.2 Create Payment Patterns 2
19.2.1 Define Absolute Payment Patterns 3
19.2.2 Define Relative Payment Patterns 5
19.2.3 Define Split Payment Patterns 8

20 User Defined Repricing Patterns

20.1 Search Repricing Pattern 1
20.2 Create Repricing Patterns 2
20.2.1 Define Absolute Repricing Patterns 3
20.2.2 Define Relative Repricing Patterns 6

21 Transfer Pricing Rules

21.1 Creating Transfer Pricing Rules 2
21.2 Defining Transfer Pricing Methodologies 3

Copyright © 2011, 2026, Oracle and/or its affiliates.

21.3 Defining Assumptions with the Default Currency 3
21.3.1 Availability of Transfer Pricing Methodologies 8
21.3.2 Required Parameters 10
21.4 Defining the Redemption Curve Methodology 11
21.4.1 Defining Tractor Methodology 11
21.4.2 Defining the Unpriced Account Methodology 12
21.5 Copying Assumptions across Currencies 12

22 Prepayment Rules

22.1 Create Prepayment Rules 1
22.2 Defining Prepayment Methodologies 2
22.2.1 Defining Assumptions with the Default Currency 2
22.2.2 Defining Constant Prepayment Method 3
22.2.3 Defining Prepayment Model Method 5
22.2.4 Defining PSA Prepayment Method 6
22.2.5 Defining the Arctangent Calculation Method 7
22.2.6 Defining Early Redemption Assumptions 8

23 Prepayment Models

23.1 Create Prepayment Models 1
23.1.1 Defining the Structure of the Prepayment Model 3
23.1.2 Prepayment Rate Calculation 3
23.1.2.1 Manual 4
23.1.2.2 Integrated Model 5
23.1.2.3 External Model 6
23.1.3 Editing Prepayment Models 7
23.1.4 Editing Prepayment Rates in a Prepayment Model 8
23.1.5 Editing Prepayment Rate Calculation 9
23.1.6 Prepayment Rate Data Loader 10
23.1.7 Prepayment Model Analysis 10
23.1.7.1 Searching for a Model Rule 13
23.1.7.2 Defining a Model 14
23.1.7.3 Editing Prepayment Model Analysis Rule 21
23.1.7.4 Deleting Prepayment Model Analysis Rule 22
23.1.7.5 Copying Prepayment Model Analysis Rule 22
23.1.7.6 Dependency Checking 22
23.1.8 Prepayment Rate Calculation Process 23
23.1.8.1 Creating a Prepayment Rate Calculation Process 24
23.1.8.2 Editing an Existing Prepayment Rate Calculation Process 26
23.1.8.3 Executing a Prepayment Rate Calculation Process 27

Copyright © 2011, 2026, Oracle and/or its affiliates.

23.1.8.4 Performance Tuning Prepayment Rate Calculation Processes 29

24 Non-Maturity Products Data Creation Process

24.1 Creating Data Creation Process 3
24.2 Executing Data Creation Process 6
24.2.1 UI Based Procedure 7
24.2.2 Batch Based Procedure 7

25 Non Maturity Products Model Analysis

25.1 Searching for a Non-Maturity Products Model Analysis Rule 2
25.2 Creating Non-Maturity Products Model 3
25.3 Editing Non-Maturity Products Model Analysis Rule 12
25.4 Deleting Non-Maturity Products Model Analysis Rule 12
25.5 Copying Non-Maturity Products Model Analysis Rule 13
25.6 Dependency Checking 13

26 Adjustment Rules

26.1 Create Adjustment Rules 1
26.2 Defining Adjustment Methods 2
26.2.1 Availability of Adjustment Methods 4
26.2.1.1 Adjustment Method Parameters 5
26.3 Define the Use TP Method from TP Rule Method 8
26.3.1 Define the Use TP Method from TP Rule Method 8
26.3.2 Define the Use TP Method from TP Rule Method 8
26.3.3 Define the Use TP Method from TP Rule Method 8

27 Stochastic Rate Indexing

27.1 Defining Stochastic Rate Indexes 1

28 Alternate Rate Output Mapping Rules

28.1 Create Alternate Rate Output Mapping Rules 2
28.1.1 Additional Steps to Create Alternate Rate Output Mapping Rule 2
28.2 Registering Alternate Output Columns for Account Tables 3

29 Propagation Patterns

29.1 Defining the Propagation Pattern 1
29.1.1 Loan Commitment Propagation 3

Copyright © 2011, 2026, Oracle and/or its affiliates.

29.2 Propagating Transfer Pricing Results 4

30 Transfer Pricing Process

30.1 Create a Standard Transfer Pricing Process 2
30.1.1 Transfer pricing for Future Dated Instruments 16
30.1.2 Implied Forward Rate Calculation 16
30.2 Creating a Stochastic Transfer Pricing Process 17
30.3 Execute a Transfer Pricing Process 30
30.3.1 Undo 31

31 Break Identification Process

31.1 Create a Break Identification Process 1
31.2 Executing a Break Identification Process 5
31.2.1 Run Rule Framework 7
31.3 Performance Tuning Break Identification Processes 7

32 Rate Locks and the Loan Commitments Pipeline

32.1 Rate Lock Options 1
32.2 Transfer Price using Forward Rates and Calculate Rate Lock Option Costs 2
32.3 Rate Management 3
32.4 Implied Forward Rates Output 5
32.5 Transfer Pricing Future Dated Instruments (Loan Commitments) 6
32.5.1 Implied Forward Rate Calculation 7
32.5.2 Option Cost Calculation 10
32.6 Execution and Results 10

33 FTP Rate Card

33.1 Setting up a Product 2
33.1.1 Defining a Product 2
33.1.2 Defining a Product: An Example 5
33.2 Setting up Batch Execution of Rate Card 7
33.3 Rate Report Templates 12
33.3.1 Search for the Rate Report Templates 13
33.3.2 Creating a Rate Report Template 13
33.3.3 Viewing or Editing a Rate Report Template 15
33.3.4 Copying an Existing Rate Report Template 15
33.3.5 Deleting an Existing Rate Report Template 16
33.4 Defining a Rate Report 17

Copyright © 2011, 2026, Oracle and/or its affiliates.

33.4.1 Defining Rate Report 18
33.4.2 Defining a Rate Report: An Example 20
33.5 Viewing Rate Card Report 22

34 Data Aggregation and Stratification

34.1 Partial Aggregation 1
34.2 Stratification Action Operations 1
34.3 Aggregation Action Operations 2
34.4 Stratification Tips 2
34.5 Setting up Stratification Definitions 3
34.5.1 Stratification Configuration 3
34.5.1.1 Setting up Stratification Tier Structures 4
34.5.2 Setting up Stratification Definitions 5
34.5.3 Defining Stratification Rules 6
34.5.4 Defining Stratification Tiers 8
34.5.5 Executing a Stratification Rule 8
34.5.5.1 Using SQL* Plus to Execute a Stratification Rule 9
34.5.5.2 Using ICC Batch Framework 10
34.5.6 Creating New Aggregate Table 10

35 Account Audit Portal

35.1 Account Attributes Tab 3
35.2 Audit Details Tab 3
35.3 Cash Flows Tab 4
35.4 Process Errors Tab 5

36 View FTP Objects in Metadata Browser

36.1 Configure Metadata Browser 2
36.2 Viewing FTP Objects in Metadata Browser 3

A Management Ledger

A.1 Statement of Direction A-1
A.2 Frequently Asked Questions A-2

B Multi-Language Support

Copyright © 2011, 2026, Oracle and/or its affiliates.

C Standard Navigation Paths

D Archive-Restore

D.1 Command Line Migration D-1
D.2 Command Line Migration Utility D-1
D.3 Supported Objects D-2
D.4 Input Values for OBJECTMIGRATION.xml D-3
D.5 Logging D-5

E Transfer Pricing Error Messages

F Non-Maturity Products Model Analysis

G Process Tuning

G.1 Defining Multiprocessing Parameters G-1
G.1.1 Assignment Levels Details G-4
G.1.2 Parameter Specification G-5
G.1.3 Multiprocessing Options G-7
G.2 Application Multiprocessing Model G-9
G.3 Multiprocessing Options G-10
G.4 Tuning Application Multiprocessing G-14
G.5 Performance Tuning Applicable to FTP G-15
G.6 Operating System-Level Tuning G-16
G.6.1 Database-Level Tuning G-16
G.6.2 Recommendations for Performance Improvement G-17

H Batch to Load the INSTRUMENT_SUMMARY_TABLE

I External Break Process Flow

J Creating and Executing T2T Processes

J.1 Load Run IDs for T2Ts J-1
J.1.1 Existing T2Ts J-2
J.1.2 Incremental T2Ts J-3
J.2 Supported Incremental T2Ts J-4

Copyright © 2011, 2026, Oracle and/or its affiliates.

J.2.1 Executing Incremental T2T J-4

K Seeded Batches

K.1 Load Instrument Summary K-2
K.1.1 Description K-2
K.2 <INFODOM>_PFT_MANAGEMENTLEDGER_LOAD_UNDO K-4
K.3 <INFODOM>_T2T_ALL_INSTRUMENT_TABLES K-4
K.4 <INFODOM>_T2T_ALL_TXN_SUMMARY K-5
K.5 <INFODOM>_T2T_MANAGEMENT_LEDGER_MULTI_SOURCE K-6
K.6 <INFODOM>_T2T_ALL_INS_SUMM_TABLES K-6
K.7 <INFODOM>_DIMENSION_ACCOUNTS K-7
K.8 <INFODOM>_T2T_FOR_FSI_D_CUSTOMER K-8
K.9 <INFODOM>_LOAD_INSTRMNT_SUMMARY K-8
K.10 List of Deprecated Instrument Summary T2Ts K-8

L FTP Web Services

L.1 Important Changes in OFSAA Release 8.1 L-2
L.2 How to test and execute the FTP Web Services? L-3
L.3 Configure the FTP Web Services L-3
L.4 Use the Web API Testing Tool L-3
L.5 Description of FTP Web Services L-4
L.6 Web Service Name: getTPRate L-4
L.6.1 Test this service using the Java command line Test Program L-5
L.6.2 Test using Web API Testing Tool (example Postman) L-6
L.7 Web Service Name: getEconomicCostWithParameters L-7
L.7.1 Input JSON format L-8
L.7.2 Test the Service using the Java Command-line Test Program L-11
L.7.3 Test using Web API Testing tool (example Postman) L-12
L.8 Web Service name: getPricing L-12
L.8.1 Test this service using the Java Command-line test program L-16
L.8.2 Test using Web API Testing tool (example Postman) L-16
L.9 Web Service name: getPricingData L-17
L.9.1 Test this service using the Java Command-line test program L-18
L.9.2 Test using Web API Testing tool (example Postman) L-18
L.10 Payment Schedule Enhanced FTP Web Service L-19
L.11 Web Service Changes for GDPR L-22
L.12 Java Programs for Testing the FTP Web Services L-22
L.13 Additional Configuration Needed for Economic Cost Web Services L-22

Copyright © 2011, 2026, Oracle and/or its affiliates.

M Glossary

Index

Copyright © 2011, 2026, Oracle and/or its affiliates.

1

Oracle Financial Services Funds Transfer Pricing

Oracle Financial Services Funds Transfer Pricing (OFS FTP) is the industry-standard software application for implementing a matched rate transfer pricing system. Recognizing the value of matched rate transfer pricing, financial institutions are increasingly incorporating it into 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.

OFS FTP calculates transfer rates at the lowest possible level of detail in your institution's balance sheet, the instrument record level. You can generate accurate charges and credits for all sources and uses of funds for your institution and measure net interest margin contribution at the instrument record level.

The following are the key benefits of Oracle Funds Transfer Pricing:

- To ensure accurate results, Oracle Funds Transfer Pricing combines advanced methodologies with a flexible and easy-to-interpret reporting approach. Oracle Funds Transfer Pricing allows you to:ul
  - Determine the account level spread earned on assets and liabilities, and the spread earned or lost as a result of interest rate risk exposure.
  - Apply multiple TP add-on rate assignments to incrementally price for liquidity risk or basis risk or incorporate pricing incentives.
  - 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.

Oracle Funds Transfer Pricing and Other Oracle Financial Services Applications

Oracle Funds Transfer Pricing operates on top of a common infrastructure, Oracle Financial Services Analytical Applications Infrastructure (OFSAAI). OFSAA applications form a comprehensive decision support solution that significantly enhances enterprise risk and performance management functions across a financial institution.

- **Oracle Financial Services Enterprise Performance Management**: More than ever, financial institutions, their regulators as well as their shareholders are focused on the need to measure and meet risk-adjusted performance objectives, price products to reflect their true risk, and better understand how their institution is impacted by threats to liquidity, capital adequacy, and exposure to market rate volatility.
- **OFSAA Data Model**: OFSAA Data Model (OFSAADM) is shipped with optimized data structures and prepackaged data elements developed specifically for the financial services

Copyright © 2011, 2026, Oracle and/or its affiliates.

industry. OFSAADM is the foundation for the OFS applications. It provides the database structures necessary to support individual business applications.

- **OFSAA Infrastructure**: OFSAA Infrastructure is the foundation for OFSAA. It provides support for User Administration, Metadata Management, a Processing Framework, a Forms Framework, and additional capabilities necessary to support the individual business applications.
- **Oracle Funds Transfer Pricing**: Oracle Financial Services Funds Transfer Pricing is the industry's first and leading matched maturity funds transfer pricing application, enabling financial institutions to determine the spread earned on assets, from funding sources, and the spread earned as a result of interest rate exposure for each and every customer relationship. This enables accurate assessment of profitability along with the product, channel, and business lines, as well as the centralization of interest rate risk so it can be effectively managed. Oracle Financial Services Funds Transfer Pricing is fully integrated with Oracle's Financial Services Analytical Applications and shares a common customer account-level data model.
  The transfer pricing process isolates the four major sources of a bank's margin:
  - The spread earned on assets
  - The spread earned on liabilities
  - The spread earned or lost as a result of interest rate exposure
  - The spread earned or lost as a result of embedded options such as prepayments

  Measuring and managing the interest margin offers tremendous profit opportunities. By separating the components of net interest income, Oracle Funds Transfer Pricing isolates rate risk into your funding center where it can be centrally managed. In turn, business units are held accountable for what they can control: pricing and profitability. Armed with this highly accurate information, companies make solid, supportable decisions that lead to increased success in the marketplace.
- **Oracle Financial Services Profitability Management**: Oracle Financial Services Profitability Management enables financial services institutions to calculate profitability by-products, channels, segments, and even individual customers. These profitability calculations are adjusted for risk, and they drive Risk-Adjusted Performance Management (RAPM), an imperative for financial services institutions operating in this rapidly evolving and complex industry.
  Isolating and analyzing business unit profitability is only the first step in building effective profitability management. To hold operational managers truly accountable for the profitability, senior management must give them the power to identify the products that are profitable in each market segment, for each customer.

  Together with OFSAA Transfer Pricing, OFSAA Profitability Management is the tool that delivers this power. Profitability Management links general ledger, account-level, and statistical data together to produce detailed financial statements for individual business units, products, and customers. Combining powerful modeling and disciplined accounting, OFSAA Profitability Management delivers complete, accurate, and inclusive profiles of profitability.
- **Oracle Financial Services Asset Liability Management**: Volatile market conditions and increasing regulatory pressures are placing greater demands on the risk management function. These stresses are driving financial institutions to review their current risk modeling and measurement capabilities. Yet, these circumstances also provide institutions with the opportunity to update technology solutions to systems fully integrated across the risk and performance spectrum. Departmental, one-off solutions are no longer viable alternatives in an environment where all systems need to work together.
  Oracle Financial Services Asset Liability Management (ALM) helps financial services institutions measure and manage interest rate risk and liquidity risk, perform balance sheet

Copyright © 2011, 2026, Oracle and/or its affiliates.

valuations, and forecast both net interest income and net income. The Oracle Financial Services ALM solution measures and models every loan, deposit, investment, and off-balance sheet instrument individually, using both deterministic and stochastic methods. This helps institutions gain a better understanding of the risks they have assumed and their sensitivity to changes in economic conditions.

- **Oracle Financial Services Balance Sheet Planning**: Oracle Financial Services Balance Sheet Planning is designed to help financial services institutions budget for a full balance sheet and the associated profit and loss statement.
  Banks have a number of very unique needs when looking ahead. They must be sensitive to economic conditions, and create plans that not only forecast future performance, but also the forward risks they are assuming. Most importantly, they require tools that accurately forecast net interest margin. By accurately modeling the detailed and complex events on a bank's balance sheet, for both the current book of business and forecasted new volumes, Oracle Financial Services Balance Sheet Planning enables the delivery of accurate margin forecast and comprehensive, meaningful budgets.
- **Oracle Financial Services Business Intelligence Applications**: OFSAA Business Intelligence (BI) Applications are complete, pre-built BI solutions that deliver intuitive, role-based intelligence for everyone in an organization—from front-line employees to senior management—that enable better decisions, actions, and business processes.

Oracle Funds Transfer Pricing Integrations

Oracle Funds Transfer Pricing integrates with the following modules:

- Oracle Financial Services Profitability Management
- Oracle Financial Services Balance Sheet Planning
- Oracle Financial Services Asset|Liability Management
- Oracle Financial Services Price Creation and Discovery

A transfer-priced balance sheet is merely the beginning. You can combine Oracle Funds Transfer Pricing results with non-interest income and expense information populated at the account level with Oracle Financial Services Profitability Management to measure total profitability based on a user-definable combination of dimensions. You can generate FTP and full risk-based pricing information in real-time with Oracle Financial Services Price Creation and Discovery. You can also integrate FTP results into Oracle Financial Services Balance Sheet Planning or Oracle Financial Services Asset Liability Management to project transfer pricing results into your forecasting and planning processes.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2

Overview of OFSAA Infrastructure

OFSAA Infrastructure includes facilities for creating and maintaining:

- Dimensional reference data
- Interest rate & currency exchange rate data
- Process tuning data

OFSAA Infrastructure also includes functionality for building and maintaining rules that may be used by any Oracle Financial Services Analytical Application. These common rule objects include:

- Expressions
- Hierarchies
- Filters

Topics:

- OFSAA Infrastructure
- Overview of Dimensionality in OFSAA
- Rollup Members for Dimensions Supporting Hierarchies
- Dimension Types
- Attributes
- OFSAA Rules
- OFSAA Filters
- OFSAA Expressions
- Folders
- Visibility of Rules
- Access Type
- Application Preferences
- Global Preferences
- System Configuraton
- Overview of Dimensionality in OFSAA
- Attributes
- Overview of OFSAA Rules
- OFSAA Filters
- OFSAA Expressions
- Folders
- Global Preferences
- System Configuration

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Application Preferences

2.1 Overview of Dimensionality in OFSAA

Dimensions are used to stratify your business data for processing or reporting purposes. For example, at a minimum General Ledger systems are typically stratified by a General Ledger dimension, an organization or cost center dimension, and a time dimension. Customer account level data will normally have these same dimensions but are often additionally segmented by dimensions such as Product or Customer. You may wish to construct models using a specialized product dimension or other customized dimensions such as geography, cost pool, transaction, activity or channel.

Dimensions are comprised of lists of members. For example, the Organizational Unit dimension might be comprised of Branch #1, Branch #2, Branch #3, etc. Some dimensions may contain tens or even hundreds of thousands of members (for example, a large organizational or customer dimension) while others may contain as few as two members (for example, a simple product-type dimension with values of Fixed and Variable).

The member values or "Chart of Accounts" for each dimension may be maintained either internally within the OFSAA Infrastructure or externally

Internally

OFSAA Infrastructure includes functionality for Adding, Viewing, Editing, or Deleting dimension member values. Which of these functions you can access depends on your logon privilege and on the dimension with which you wish to work. For details, on the creation and maintenance of Dimension Members, refer to the Oracle Financial Services Analytical Applications Infrastructure User guide.

Externally

You may choose to source and maintain some Charts of Accounts from external systems such as your General Ledger, a Data Warehouse, or from some other external master data management system. For more information, see the Oracle Financial Services Analytical Applications Data Model Utilities User Guide.

Combination

You may also choose to maintain some Chart of Account values from an external source while maintaining other Chart of Account values internally.

- Referential Integrity
- Dimension Attributes
- Hierarchies
- Rollup Members for Dimensions Supporting Hierarchies
- Dimension Types

2.1.1 Referential Integrity

You may have dimension member values defined in your Chart of Accounts that do not exist in your business data, particularly if the dimension member values are used as attribute values (see discussion of attributes below). But you would never want to have an occurrence of a dimension member value in your business data that did not exist in the dimension's Chart of Accounts. When you load business data into OFSAA, you need assurance that there are no dimension member values present in the business data that are not defined in your Chart of

Copyright © 2011, 2026, Oracle and/or its affiliates.

Accounts. For details on ensuring referential integrity between your business data and your Chart of Accounts data, see the Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration Guide.

2.1.2 Dimension Attributes

Some OFSAA dimensions support attributes. Dimension attribute values are used to qualify dimension members. For example, a cost center, say Branch #1, might have a Headcount attribute with a value of 32 and a Volume attribute with a value of High. A product dimension member, say 30-year fixed-rate jumbo LIBOR mortgage, might have a Rate attribute with a value of LIBOR, a Size attribute with a value of Jumbo, and a Loan Type attribute of Fixed. A General Ledger account, say Benefit Expense, might have an Account Type attribute with a value of Non-Interest Expense.

Like dimensions, attributes are used to stratify your business data for processing or reporting purposes. Some dimension attributes are seeded within the OFSAA data model and are used extensively in processing. You may define as many user-defined attributes as you need.

OFSAA Infrastructure includes functionality for Adding, Viewing, Editing, or Deleting dimension attributes. Which of these functions you can access depends on your logon privilege and on the dimension with which you wish to work.

Once an attribute has been defined for a dimension, attribute values may be assigned to individual dimension member values. These attribute values may be open-ended (dates, numbers, or strings) or may be restricted to a defined set of members.

2.1.3 Hierarchies

Some OFSAA dimensions support hierarchies. Hierarchies may be used to provide sophisticated stratification for either processing or reporting purposes. For example, an organizational hierarchy might start with a Division level containing Western Region, Eastern Region, and Southern Region; the next level down within the hierarchy might be state or county. A product hierarchy might begin with branches for Asset vs. Liability vs. Service products; under the Asset branch, you might define additional branches for Mortgage Lending, Commercial Lending, Consumer Lending, and so on. Hierarchies are used extensively in OFSAA models to assign methods to products and to support allocation methodologies.

Like dimension members, dimension hierarchies may be maintained either internally within the OFSAA Infrastructure or externally.

Internally

OFSAA Infrastructure includes functionality for Adding, Viewing, Editing, or Deleting hierarchies. Which of these functions you can access depends on your logon privilege and on the hierarchy with which you wish to work.

Externally

You may choose to source and maintain some or all of your hierarchies from external systems such as your General Ledger, a Data Warehouse, or from some other external master data management system, such as Oracle Data Relationship Management (DRM). For more information on loading hierarchy data from external systems, see the Oracle Financial Services Analytical Applications Data Model Utilities User Guide.

Combination

You may also choose to maintain some hierarchies from an external source while maintaining other hierarchies internally.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2.1.4 Rollup Members for Dimensions Supporting Hierarchies

For dimensions that support hierarchies, every dimension member value has the property of being either a **leaf** value (child-only or last descendent child value) or a **rollup node** value. For example, if Branch #1, Branch #2, and Branch #3 all belonged to the West rollup point of an organizational hierarchy, then West would also be a dimension member value of the Organizational Unit dimension. Your business data is typically expressed at the leaf level.

2.1.5 Dimension Types

OFSAA supports the following three fundamentally different kinds of dimensions:

- Key Processing Dimensions
- Standard Dimensions
- Simple Dimensions

- Key Processing Dimensions
- Standard Dimensions
- Simple Dimensions

2.1.5.1 Key Processing Dimensions

Key Processing Dimensions have the following features:

- Accessible as modeling dimensions for all of the OFSAA analytical engines.
- Expressed as columns in nearly all of your business fact tables.
- Support both attributes and hierarchies

Metadata for Key Processing Dimensions are stored in four tables:

- A base table (For example, DIM_PRODUCTS_B)
- A translation table (For example, DIM_PRODUCTS_TL)
- An attribute table (For example, DIM_PRODUCTS_ATTR)
- A hierarchy table (For example, DIM_PRODUCTS_HIER)

Base tables store basic information about each dimension member and translation tables store names and descriptions for each dimension member in multiple languages. Attribute tables store one or many attribute values for each dimension member. Hierarchy tables store one or more hierarchies for each dimension (you may define as many hierarchies as you wish for any dimension that supports hierarchies).

Seeded Key Processing Dimensions

OFSAA is seeded with six Key Processing dimensions:

- Financial Element
- Organizational Unit
- General Ledger Account
- Common Chart of Accounts
- Product

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Legal Entity

Among the six seeded Key Processing dimensions, only the Financial Element dimension is pre-populated with a Chart of Accounts, but it is only populated in the range between 0 and 10,000 (For more details on the seeded Chart of Accounts for the Financial Element dimension see the Oracle Financial Services Profitability Management User Guide. This is a reserved range in which you may not add new members, or edit or delete existing members. You may add user-defined Financial Elements anywhere outside the reserved range.

User-Defined Key Processing Dimensions

For more information on adding your own user-defined Key Processing dimensions, see the Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration Guide.

2.1.5.2 Standard Dimensions

Standard dimensions may support attributes and/or hierarchies depending on how they are configured, but are not used as processing dimensions within any of the analytical application engines. Like Key Processing Dimensions, metadata for Standard Dimensions is stored in a base table and in a translation table. A Standard Dimension may or may not have an attribute table or a hierarchy table depending it is configured. Within the OFSAA modules, Standard dimensions that are defined as having a hierarchy table will support hierarchies, and Standard dimensions that are defined as having an attribute table will support attributes. See the Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration Guide.

Seeded Standard Dimensions

OFSAA is seeded with two Standard Dimensions

- Country
- Customer

Both dimensions support attributes and hierarchies. Default member values are provided for the Country dimension, but no values are provided for the Customer dimension.

User-Defined Standard Dimensions

Refer to the Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration Guide for details on adding user-defined Standard dimensions.

2.1.5.3 Simple Dimensions

Sometimes referred to as Code dimensions, Simple dimensions are a "list of values" that support neither attributes nor hierarchies. Their three key uses are:

- Reserved for use within the analytical application engines
- Stratifying your data for process or report filtering purposes
- Serving as the **list of values** when used as attributes

The OFSAA Data Model comes with more than 150 seeded Simple dimensions. Many of these seeded Simple dimensions are also seeded with dimension members. Unlike Key Processing dimensions and Standard dimensions, Simple dimensions are stored in two tables:

- A code table (For Example, AMRT_TYPE_CD)
- A Multi-Language Support or MLS table (For Example, AMRT_TYPE_MLS)

Copyright © 2011, 2026, Oracle and/or its affiliates.

Simple Dimension "Process Identifier Code" helps to improve ALM process execution performance by skipping VALIDATE_NODE_MAP_ALM package execution incase there is no change in underlying metadata while running ALM Process everytime. There is no direct method to identify whether any change in underlying data at the node level and one need to execute VALIDATE_NODE_MAP_ALM and thats where the "Process Identifier Code" Simple Dimension takes place. This dimension should contain the list of process Ids for which there is no change in underlying node level metadata hence no need to run VALIDATE_NODE_MAP_ALM everytime the process is run. In turn whenever ALM Engine is run; it checks the currently considered process id and checks its existence in FSI_PROCESS_IDENTIFIER_CD table; if present then ignores VALIDATE_NODE_MAP_ALM procedure run.

- Reserved Simple Dimensions
- Simple Dimensions Used for Processing or Reporting Purposes
- Simple Dimensions Used as Attribute Domains

2.1.5.3.1 Reserved Simple Dimensions

Accrual Basis Code is an example of a seeded Simple dimension that is used extensively by the application modules. It is seeded with a small list of values (such as 30/360, Actual/360, Actual/Actual, and so on.) and its value set may not be modified.

Amortization Type Code is another example of a reserved Simple Dimension. Amortization Type Code is populated with reserved values that the Cash Flow engine interprets, and its value set may not be modified. Many other Simple Dimensions are reserved and may not be modified.

2.1.5.3.2 Simple Dimensions Used for Processing or Reporting Purposes

This kind of Simple Dimension cannot be used directly by the application engines, but they are accessible to OFSAA Filter rules.

Credit Rating is an example of a seeded Simple Dimension that may be used for processing or reporting purposes. You may, for example, wish to allocate risk equity to individual instruments as a function of Credit Rating. The seeded values for Credit Rating are Moody's credit ratings, but you are free to completely redefine the membership of this dimension. Geographic Location is another example. Geographic Location is pre-populated with the US States and Canadian provinces, but you are free to completely redefine the completely geographic locations.

2.1.5.3.3 Simple Dimensions Used as Attribute Domains

Both seeded and user-defined attributes often utilize a Simple Dimension to provide a list of values. For example, you might create an Adjustable Type attribute on your Product dimension and you might want this attribute to be limited to the values of Fixed, Variable, or Administered. In this case, you might first build a Simple Dimension called Type whose dimension members were limited to three values: Fixed, Variable, or Administered. Next you could assign the Type dimension to be the domain of your new Adjustable Type attribute.

See Oracle Financial Services Analytical Applications Infrastructure (OFSAAI) Installation and Configuration Guide.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2.2 Attributes

OFSAA Infrastructure includes functionality for Adding, Viewing, Editing, or Deleting dimension attributes. Which of these functions you can access depends on your logon privilege and on the dimension with which you wish to work.

The seeded OFSAA Key Processing Dimensions have a number of seeded or seeded & required attributes. For example, the Common Chart of Accounts dimension includes an Account Type attribute and an Accrual Basis attribute; the Financial Element dimension includes an Aggregation Method attribute. Some of these attributes have important implications in processing within OFSAA.

You may add Attributes to any Key Processing Dimension or to any Standard Dimension that supports Attributes (see sections above describing Key Processing Dimensions and Standard Dimensions).

One of the important properties of an Attribute is its Data Type. OFSAA Infrastructure supports Dimension Attribute Data Types of:

- Date, String, or Number
- Dimension
- Unbounded and Bounded Attribute Data Types

2.2.1 Unbounded and Bounded Attribute Data Types

Dimension attribute data types may be either bounded or unbounded. The Date, String, and Number types are unbounded, that is, there is no predetermined set of attribute values. A Date attribute value may be any valid date, a String attribute value may be any string up to 30 characters in length, and a Number attribute value may be any number. Examples of these kinds of attribute data types might include:

- Cost Center Open Date (a Date attribute on the Organizational Unit dimension)
- Product Memo (a String attribute on the Product dimension)
- Cost Center Headcount (a Number attribute on the Organizational Unit dimension)

Unlike attributes having the Date, String, or Number data type, attributes whose data type is Dimension have defined sets of values. For example, the Product dimension has a seeded attribute called Accrual Basis the data type of which is Dimension. The Accrual Basis dimension is seeded with 8 values. Accrual Basis is a restricted dimension (you may View Accrual Basis dimension members but you may not delete them or add new accrual bases), but whenever you assign an Accrual Basis attribute value to one of your Product dimension member values, you must select from the list of 8 available Accrual Bases.

2.3 Overview of OFSAA Rules

OFS Analytical Application models are constructed using combinations of OFSAA rules. For example:

- An Allocation rule might utilize a Hierarchy Filter rule, an Expression rule, and a Static Table Driver rule
- A Transfer Pricing rule might utilize a Hierarchy rule, a Data Element Filter rule, and a Prepayment rule

Copyright © 2011, 2026, Oracle and/or its affiliates.

- A Data Element Filter rule might utilize an Expression rule
- A Group Filter rule always incorporates one or more underlying Data Element Filter rules
- Hierarchy Filter rules always utilize an underlying Hierarchy rule

The advantage of constructing models from discrete rule components is that individual rules may be re-used dozens or even hundreds of times. For example, if you wish to build a large series of allocation rules which operate on non-interest expense data in the Western region, you may build filters for (1) non-interest expense and (2) the Western region and then utilize these filters in every allocation rule. As another example, if you wanted to assign the same transfer pricing method and assumptions to all mortgage products, you could build a Hierarchy on the product dimension containing a Mortgage Products rollup point and assign your method and assumptions at the rollup level. You could subsequently utilize the same Hierarchy rule for allocation processing or reporting purposes.

Many rule types are specific to individual analytical applications (For example, Transfer Pricing rules, Cash Flow Engine rules, Allocation rules, and so on.) while OFSAA Infrastructure rules are shared across all OFS applications. Rule types that are shared across all applications include:

- Hierarchies
- Hierarchy Filters
- Data Element Filters
- Group Filters
- Expressions

Dimensions, dimension members, attributes, and currency exchange rates are also shared across all applications.

- Dependencies

2.3.1 Dependencies

OFSAA rules often depend on other rules for some or all of their data or usefulness. For example, an Allocation rule may depend on a Data Element Filter rule to select and focus data in a meaningful way. If that Data Element Filter is deleted, then the Allocation rule becomes unusable. OFSAA applications track such dependencies and ensure that users may not delete rules that are consumed by other rules. Additionally, rule summary screens that present lists of rules that you have defined, offer functionality for you to generate reports on rule interdependency.

2.4 OFSAA Filters

OFSAA Infrastructure supports following filters:

- Data Element Filters
- Group Filters
- Hierarchy Filters
- Attribute Filters

Data Element Filters

Data Element Filters are used to select and focus data. For example, you may transfer price some financial instruments at their source and may therefore wish to ensure that you do not

Copyright © 2011, 2026, Oracle and/or its affiliates.

overwrite the transfer rates for any instruments that already have a non-zero transfer rate. In such a case, you could construct a Data Element Filter to isolate only those instruments having a non-zero transfer rate. As another example, you may wish to allocate a charge to demand deposit accounts having small balances. In this case, you could construct a Data Element Filter to focus on just those instrument records having balances less than your chosen threshold.

Group Filters

Group Filters are used to join together one or more Data Element Filters. For example, if you built one Data Element Filter that focused on balances less than 10,000 and a second Data Element Filter that focused on accounts whose origination date was prior to 2009, a Group Filter built on the two Data Element Filters would focus on accounts with balances less than 10,000 AND whose origination dates were prior to 2009.

Hierarchy Filters

Hierarchy Filters allow you to focus on data that rolls up to one or more points within a chosen hierarchy. You may also focus on "leaf" level data or a combination of "leaf" level and rollup levels. For example, if your organization had a Western, Northern, Southern, and Eastern division, you could construct a Hierarchy Filter that focused on data belonging to branch 1, 2, or 7 within the Eastern division plus any data belonging to the Northern division.

Attribute Filters

Attribute Filters allows you to use Attribute of a dimension, as a filter condition to select data for processing. For example, you may want to filter and select only those products for processing which has Product Attribute of ‘Amenable Category’ mapped as ‘Amenable’.

2.5 OFSAA Expressions

Expressions represent formulas for combining data from your data model (For example, Column A divided by Column B times Column C). The results of Expressions may be invoked within a number of OFSAA rule types (For example, Data Element Filters, Allocation rules, and so on).

2.6 Folders

One of the fundamental properties of any OFSAA rule is the Folder in which it is stored. Your System Administrator will configure your user ID to have access to one or more folders in which you can store your rules. Your access rights will likely be similar to co-workers who have a similar business role. For details on security management and defining Segment Maintenance in OFSAA, refer to Oracle Financial Services Analytical Applications Infrastructure User Guide.

Storing rules in folders gives you the means to segregate your rules for functional or security considerations. For example, you may wish to:

- Set up folders to store Transfer Pricing rules that are separate from folders you have built for Profitability Management rules and that are also separate from folders you have built for Asset/Liability Management rules
- Establish some folders for rules you are in the process of developing and testing and other folders for fully tested, production-ready rules

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Segregate some rules that are fundamental to the definition of other rules (For example, some Data Element Filters may be used in dozens or hundreds of other rules) to ensure that access to those fundamental rules is highly controlled
- Segregate your hierarchies, which are also frequently fundamental to OFSAA models, into a restricted-access folder
- Visibility of Rules
- Access Type

2.6.1 Visibility of Rules

Every navigation path to view a rule in OFSAA begins with a Summary screen. Within the Summary screen for each rule type, a user may browse through all of the rules that are stored within Folders to which the user has access. From each Summary screen, users may be granted the ability to Add, Edit, View, or Delete rules.

When building rules in an OFS Analytical Application, you frequently call upon other OFSAA rules (For example., a Data Element Filter nested within an Allocation rule or a Hierarchy utilized within a Transfer Pricing rule). When nesting a child rule within a parent rule you must specify the child rule by its Name and Folder (where both Name and Folder are presented in drop-down list boxes). In this context, the Folder drop-down list box for every user will include every Folder in the Information Domain regardless of his Folder access rights. This means that a user may invoke a child rule from any Folder even if he cannot access that child rule directly through its Summary screen.

Additionally, whenever a user nests a parent rule within a child rule, the OFSAA user interface provides the user the ability to view a read-only version of the underlying child rule.

2.6.2 Access Type

Another fundamental property of every OFSAA rule is its Access Type.

Access Type may be set to either Read/Write or Read Only. Only the "creator" or "owner " of a rule may modify the rule if it is set to Read Only. The user ID under which a rule is initially built is the owner of a rule. You may modify a rule that you do not own provided that:

- The rule has been stored with the Read/Write access type
- You have been granted Edit privilege for the rule's type
- You have been granted access to the Folder in which the rule is stored

Again, for details on security management in OFSAA, refer Oracle Financial Services Cash Flow Engine Installation and Configuration Guide .

2.7 Global Preferences

Clicking Global Preferences invokes the Global Preferences screen. Global Preferences items are used to configure your user interface. Here, you set your default application language and date format, and your desired Pagination Records. Pagination Records determine how many rows will be displayed on summary screens. If you select Pagination Records to be 25 records, then any screen displaying results in a tabular format will display a maximum of 25 records. To see the next set of 25 records, you can use the Next VCR button.

Functional Currency and Signage are installation-time parameters that may not be reset in Global Preferences.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Functional Currency is defined as the currency of the primary economic environment in which an entity conducts its business. For details on signage, see the Oracle Financial Services Analytical Applications Profitability Management User Guide.

Normally, you will set preferences for yourself, but if you are logged in as an Administrator, you may set Global Preferences for All Users. When setting preferences for All Users, you may restrict the ability of non-Administrators to change a Global Preference item by deselecting the Is Editable check box. If a preference item is defined as not editable, a user who is not an Administrator will inherit the value of the preference item that his Administrator has set for him and he will not have the ability to change it for himself.

2.8 System Configuration

Once an application pack is installed, you can use only the Production or Sandbox information domain, created during the installation process. Though there is an option to create a new Information Domain, there is no menu to work with the frameworks on this new information domain. This new information domain then created acts only as a Sandbox Infodom. The Create New Application feature allows you (System Administrator) to create a new Application other than the standard OFSAA Applications and associate the standard/default platform framework menu with it, thereby enabling the new application for usage. The standard platform framework menu is seeded and rendered.

For more information, refer to System Configuration section in Oracle Financial Services Analytical Applications User Guide

2.9 Application Preferences

In addition to Global Preferences, each of the OFS Analytical Applications has its own local application preferences. In the same way as in Global Preferences, Application Preferences (for each application) employs a "delegation" model that allows Administrators to set some preference items for all users while allowing non-administrative users to personalize other preference items.

Normally, you will set Application Preferences for yourself, but if you are logged in as an Administrator, you may set Application Preferences for all users of an application. When setting preferences for all users, you may restrict the ability of non-Administrators to change an Application Preference item by deselecting the **Is Editable** check box. If a preference item for a non-administrative user is defined as **not editable**, a user who is not an Administrator will inherit the value of the preference item that his Administrator has set for him and he will not have the ability to change it for himself.

While most Application Preferences items are unique to the application to which they belong, two properties are common to Application Preferences for each of the OFS Analytical Applications: Folder Name and Access Type.

Having the Folder Name and Access Type properties set at the application level allows you to have a different default Folder and Access Type for each OFS Analytical Application. Within the Left Hand Side or LHS menu, all OFSAA Infrastructure rules (Hierarchies, Filters, and Expressions) are maintained under Master Maintenance.

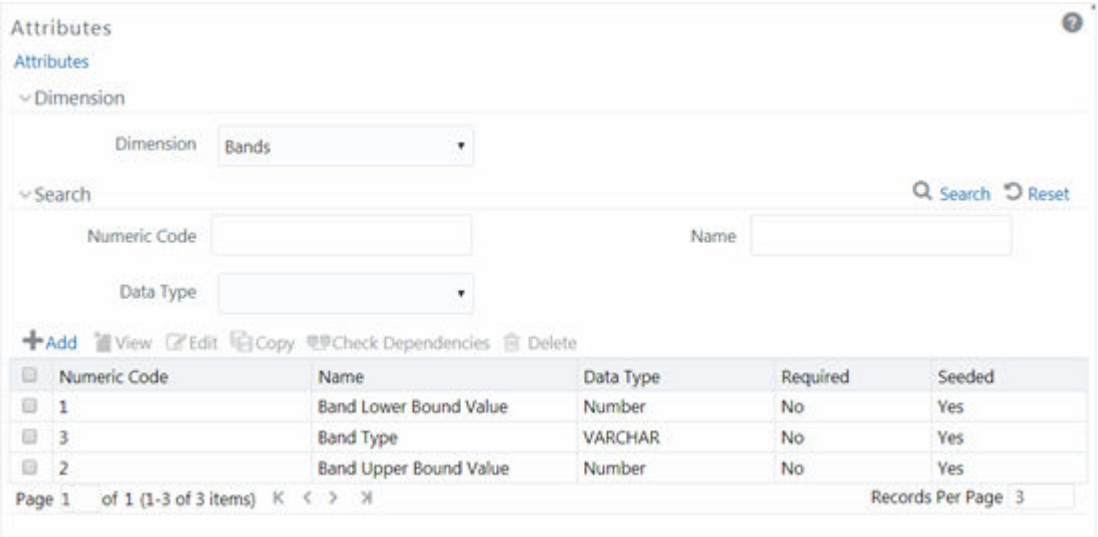
Since Folder and Access Type are not Global Preference items, a user must select one application's set of Application Preferences to be **Active** to govern the default values of Folder and Access Type for Oracle Financial Services Analytical Applications Infrastructure rules. This is done by selecting one of your Application Preference sets to be the one that is **Active for Master Maintenance**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

3
OFSAA Attributes

Attributes refers to the distinguished properties or qualifiers that describes a dimension member. Attributes may or may not exist for a simple dimension. Attributes section is available within the Dimension Management section of Financial Services Applications module.

Figure 3-1 Attributes window



The Attributes window displays the list of pre-defined Dimension Attributes with the other details such as the Numeric Code, Name, Data Type, Required, and Seeded. You can search for a specific Attribute based on Numeric Code, Name, or Data Type and view the list of existing definitions within the system.

- Adding Attribute Definition
  Attributes facilitates you to define the properties or qualifiers for the Dimension members. The Write role should be mapped to your user group, from the User Group Role Map window.
- Attribute Definition Operations
  You can perform the following operations on the Attribute Definitions:

3.1 Adding Attribute Definition

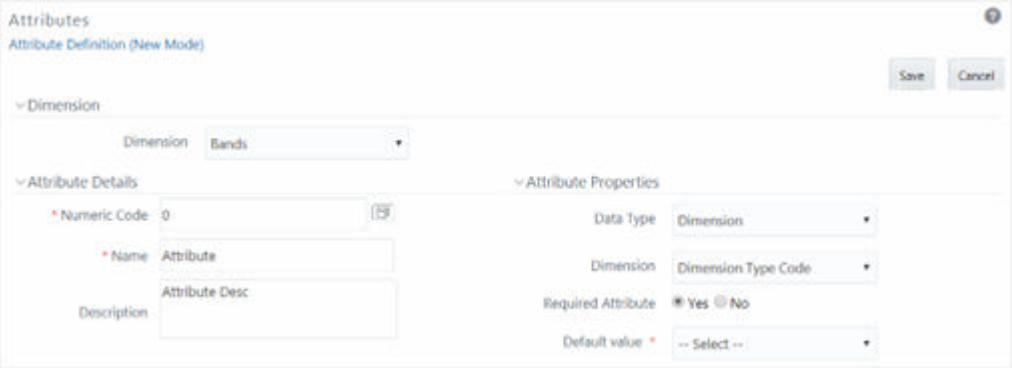
Attributes facilitates you to define the properties or qualifiers for the Dimension members. The Write role should be mapped to your user group, from the User Group Role Map window.

To create an Attribute definition in the Attributes window:

1. From the Attributes window, click **Add**. The Attribute Definition (New Mode) window is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 3-2 Attributes window



2. In the Dimension section, select the required dimension from the drop-down list.
3. Click the button next to the Numeric Code field. A unique code is auto generated. You can also manually enter the code in the Numeric Code field.
4. Enter the **Name** and required **Description** for the Attribute.

> **Note**
>
> - **Name**: The characters & ' " are restricted in the name field.
> - **Description**: The characters ~&+' "@ are restricted in the description field.

5. Enter the following details:

> **Note**
>
> Fields marked in red asterisk (*) are mandatory.

Table 3-1 Fields in the Attributes window and their Descriptions

Field	Description
Type	Select the **Data Type** as DATE, DIMENSION, NUMBER, or STRING from the drop-down list. If **NUMBER** is selected as the Data Type: The **Scale** field is enabled with "0" as default value. Enter a Scale value >= 0. If it is left as 0, values for this attribute will be limited to Integers. If you wish to enable decimal entries for this attribute, the maximum Scale value must be > 0 and <= the scale defined for NUMBER_ASSIGN_VALUE in the dimension's underlying attribute table. See the Data Model Utilities Guide for further details on the attribute table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 3-1 (Cont.) Fields in the Attributes window and their Descriptions

Field	Description
Required Attribute	Select **Yes** or **No**. If this is set to No, an attribute value is optional for the associated dimension members. **NOTE**: This field is disabled in Add and Edit modes if any members already exist for the Dimension upon which this attribute is defined.
Default Value	If **Required Attribute** is set to Yes, a Default Value must be entered, otherwise it is optional. If **DIMENSION** is selected as the Data Type: Select the **Default Value** from the drop-down list of members mapped to the selected Dimension. If the required Member is not listed in the drop-down then select --**More**—and the Member Search window is displayed. If **NUMBER** is selected as the Data Type: Enter a numeric value in the Default Value field, and it must be consistent with the Scale you have defined. If **DATE** is selected as the Data Type: Click the **Calendar** button to select a valid date as the Default Value from the calendar. If **STRING** is selected as the **Data Type**: Enter alphanumeric value in the **Default Value** field. The Maximum characters allowed in Default value field for String Data Type is 1000.

6. Click **Save**. The entries are validated and the defined Attribute is captured.

3.2 Attribute Definition Operations

You can perform the following operations on the Attribute Definitions:

Viewing Attribute Definition

You can view individual Attribute Definition details at any given point. The Read only role should be mapped to your user group.

To view the existing Attribute Definition details in the Attribute window:

1. Select the checkbox adjacent to the Numeric Code of the Attribute, whose details are to be viewed.
2. Click the **View** button in the Dimension Attributes tool bar.
   The View–Attributes window is displayed with the details such as Dimension, Numeric Code, Name, Description, and Attribute Properties.

Modifying Attribute Definition

You can modify the Name, Description, or Default Value fields of an attribute definition. The Write role should be mapped to your user group.

To modify an existing Attribute Definition in the Attributes window:

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. Select the checkbox adjacent to the Numeric Code of the Attribute, whose details are to be updated.
2. Click the **Edit** button in the Dimension Attribute tool bar. The Edit button is disabled if you have selected multiple Attributes. The Edit - Attributes window is displayed.
3. Edit the Attribute details such as Name, Description, or Default value.
   For more information, see Adding Attribute Definition.
4. Click **Save**.

Copying Attribute Definition

The Copy Attribute Definition facilitates you to quickly create a new Attribute Definition based on the existing attributes or by updating the values of the required attributes. The Write role should be mapped to your user group.

To copy an existing Attribute Definition in the Attributes window:

1. Select the checkbox adjacent to the Numeric Code of the Attribute, whose details are to be duplicated.
2. Click the **Copy** button in the Dimension Attributes toolbar to copy a selected Attribute definition. Copy button is disabled if you have selected multiple Attributes.
3. In the Copy – Attributes window you can:
   - Create new attribute definition with existing variables. Specify new Numeric Code and Attribute Name.
   - Create new attribute definition by updating the required variables. Specify new Numeric Code and Attribute Name. Update the required details. For more information, see Adding Attribute Definition.
4. Click **Save**.
   The new attribute definition details are displayed in the Attributes window.

Attribute Definition Dependencies

You can view the dependencies of Attributes. The Read only role should be mapped to your user group.

To view the dependency of attribute in the Attributes window:

1. Select the checkbox adjacent to the Numeric Code of the Attribute whose dependency is to be checked.
2. Click **Check Dependencies** button in the Dimension Attributes toolbar. The Check Dependencies button is disabled if you have selected multiple attributes.
   The Attributes Dependency Information window is displayed with the dependency details.

Deleting Attribute Definition

You can remove the Attribute Definitions which are not required in the system by deleting from the Attributes window. The Write role should be mapped to your user group.

1. Select the checkbox adjacent to the Numeric Code(s) of the Attributes whose details are to be removed.
2. Click Delete button in the Dimension Attributes tool bar.
3. Click OK in the information dialog to confirm deletion.

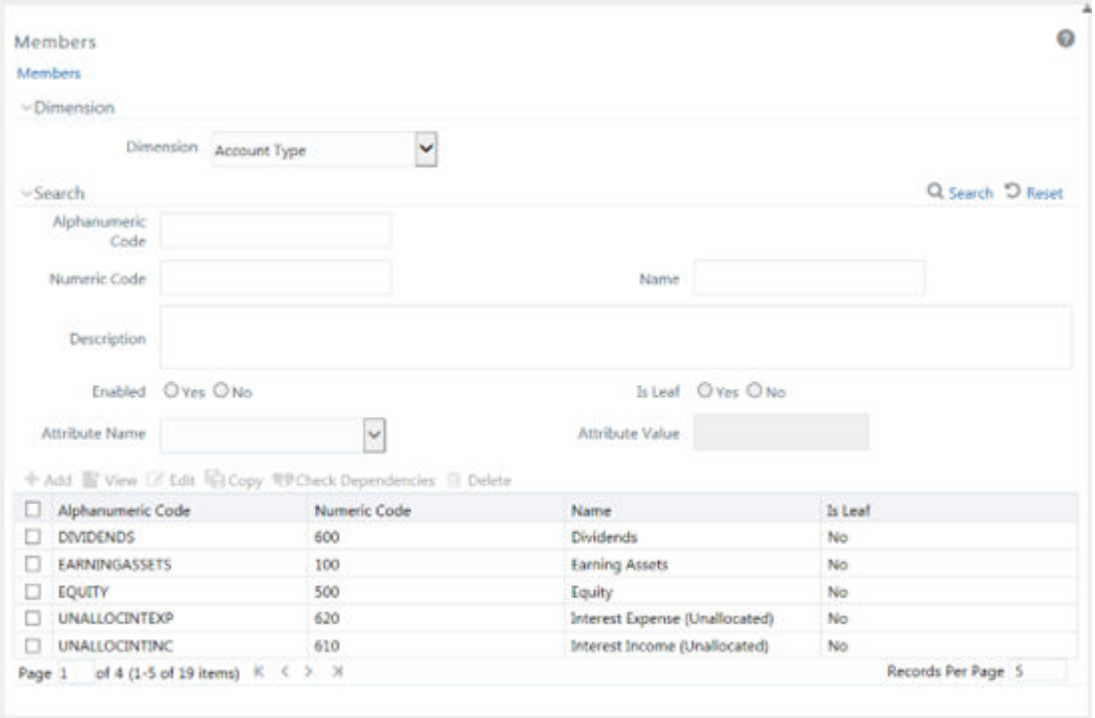
Copyright © 2011, 2026, Oracle and/or its affiliates.

4
Dimension Members

Dimension Members refer to the individual items that constitute a dimension when data is categorized into a single object. Example, Product, Organization, Time, and so on. Members are available within Dimension Management section of the Infrastructure system.

For more information on how to set up alphanumeric and numeric codes, see Configurations to use Alphanumeric and Numeric Codes for Dimension Members section in OFSAAI Administration Guide.

Figure 4-1 Members window



The Members window displays the list of pre-defined Dimension Members with the other details such as the Alphanumeric Code, Numeric Code, Name, and Is Leaf. You can also search for a specific Member based on Alphanumeric / Numeric Code (irrespective of whether dimension is configured to be numeric or alphanumeric), Name, Description, Enabled status, Is Leaf status, Attribute Name, or Attribute Value and view the list of existing definitions within the system.

- Adding Member Definition
  This option allows you to add member definition. The Write role should be mapped to your user group.
- Viewing Member Definition
  You can view individual Member Definition details at any given point.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Modifying Member Definition
  To modify an existing Member Definition in the Members window:
- Copying Member Definition
  The Copy Member Definition facilitates you to quickly create a new Member Definition based on the existing attributes or by updating the values of the required members.
- Member Definition Dependencies
  You can view the dependencies of Members.
- Deleting Member Definition
  You cannot delete predefined members or the members which are the Nodes for a hierarchy.

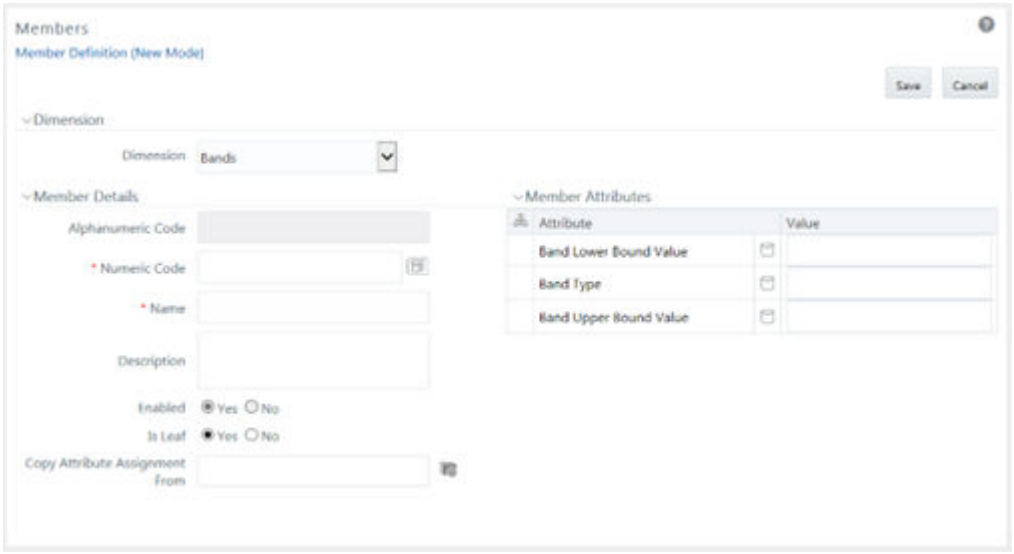
4.1 Adding Member Definition

This option allows you to add member definition. The Write role should be mapped to your user group.

To create an Attribute definition in the Attributes window:

1. Click **Add** from the toolbar. The Member Definition (New Mode) window is displayed.

Figure 4-2 Members Add window



2. In the Dimensions section, select the required Dimension from the drop-down list.
3. Enter the following Member Details:

Table 4-1 Fields in the Members Add window Field and their Descriptions

Field	Description
Fields marked in red asterisk (*) are mandatory.	

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 4-1 (Cont.) Fields in the Members Add window Field and their Descriptions

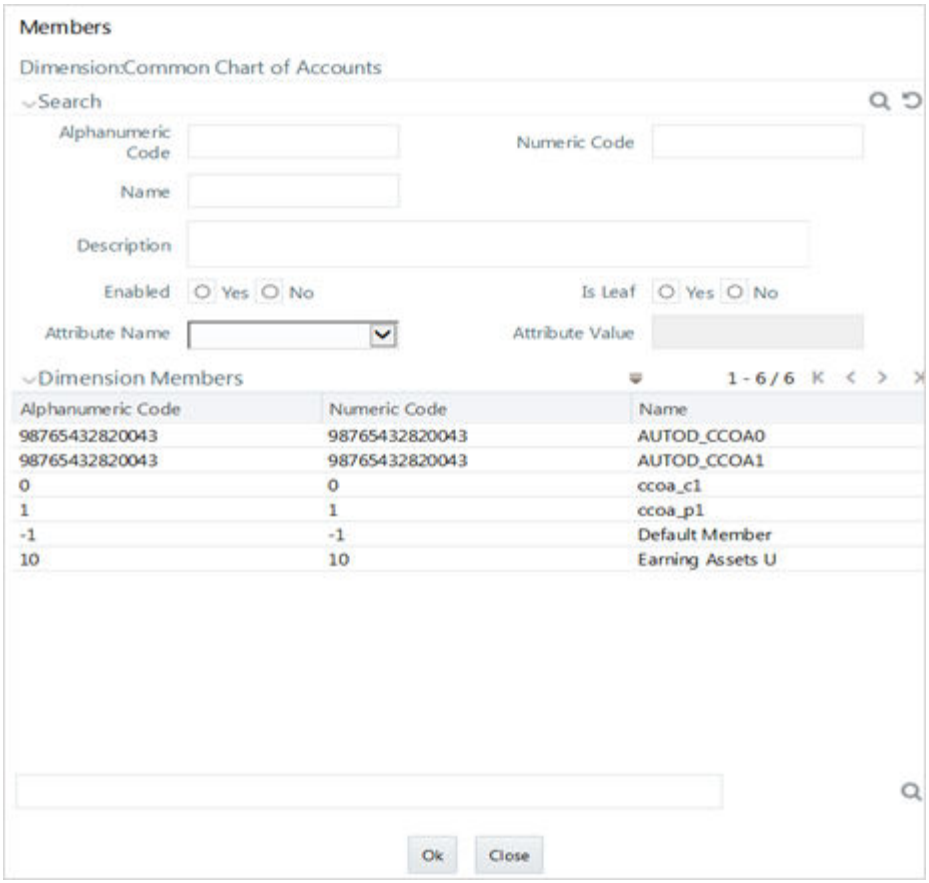
Field	Description
Alphanumeric Code	The **Alphanumeric Code** field is editable only if the selected Dimension accepts Alphanumeric Code. For example, **Billing Method** Dimension. Else, the field is Read Only and the value is fetched from the Numeric Code field entered. Enter the required Alphanumeric Code. Ensure that the code has a maximum of 14 characters and there are no special characters like & ' ~ " @ + included.
Numeric Code	Enter the Numeric Code by doing any of the following: To auto-generate a Numeric Code, click the button next to the field. A system generated code is displayed. Manually enter the required code which is auto validated for uniqueness. A maximum of 14 numeric characters can be specified. Note: if the selected Dimension accepts only Numeric Code, then the specified, the Numeric Code is auto populated to the **Alphanumeric Code** field also.
Name	Enter the Name of the Member. **Note**: The characters &' " are restricted
Description	Enter the required Description for the Member. **Note**: The characters ~&+' "@ are restricted.
Enabled	This field is set to Yes by default and is editable only in Edit window. **Note**: You can change the option to No only when the particular member is not used in any hierarchy. The disabled members will not be displayed in Hierarchy rules, or UIs which are based on Hierarchies, such as Hierarchy Filters and hierarchical assumption browsers used in applications.
Is Leaf	This field is set to Yes by default. If **Yes**, the particular member can be used as a leaf node in any hierarchy and child cannot be added to this node. If **No**, the node becomes a non-leaf and can have child nodes. **Note**: A member created as Non Leaf having child nodes to it in any hierarchy cannot be made Leaf.

> **Note**
>
> If the Dimension is selected as “Common Chart of Accounts”, proceed further. Else, jump to step 5.

4. Click **Browser** button in **Copy Attribute Assignment From** field. The Member Browser Properties window is displayed. This field can be left blank so that the Member Attributes panel can be filled in without considering the values already assigned.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 4-3 Member Browser Properties window



- Select the required Member from the Dimension Members list.
  Click the **Search** button in the Search grid to search for a specific Member based on Alphanumeric Code, Numeric Code, Name, Description, Enabled status, Is Leaf status, Attribute Name, or Attribute Value. You can also click the **Search** button to find a member present in the Dimension Members grid using key words.
- Click **OK**. The selected Member is displayed in the **Copy Attribute Assignment From** field in New – Member Details window and the details of selected Attribute are displayed in the Member Attributes section. You can edit the Attribute details as indicated:
  - Edit Attribute based on date by clicking the **Calendar** icon.
  - Edit Attribute based on Dimension Value by selecting from the drop-down list.
  - Edit Attribute based on Number Value by entering the valid numerical value.
  - Edit Attribute based on String Value by specifying alphanumerical value.

5. Click **Save** and the defined Member Definition is captured after validating the entries.

4.2 Viewing Member Definition

You can view individual Member Definition details at any given point.

To view the existing Member Definition details in the Members window:

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. Select the checkbox adjacent to the Alphanumeric Code of the Member, whose details are to be viewed.
2. Click the **View** button in the toolbar.
   The View – Member Details window is displayed with the details such as Dimension, Member Details, and Member Attributes details.

4.3 Modifying Member Definition

To modify an existing Member Definition in the Members window:

1. Select the checkbox adjacent to the Alphanumeric Code of the Member, whose details are to be updated.
2. Click **Edit** button in the toolbar. **Edit** button is disabled if you have selected multiple Members. The Edit – Member Details window is displayed.
3. Edit the Member details as required. For more information, see Add Member Definition.
4. Click **Save** to save the changes.

4.4 Copying Member Definition

The Copy Member Definition facilitates you to quickly create a new Member Definition based on the existing attributes or by updating the values of the required members.

To copy an existing Member Definition in the Members window:

1. Select the checkbox adjacent to the Alphanumeric Code of the Member, whose details are to be duplicated.
2. Click the **Copy** button in the toolbar to copy a selected Member definition. **Copy** button is disabled if you have selected multiple Members.
3. In the Copy – Member Details window you can:
   - Create new Member with existing variables. Specify the Numeric Code and new Member Name.
   - Create new Member definition by updating the required variables. Specify the Numeric Code and new Member Name.
4. Update the required details. For more information, see Add Member Definition.
5. Click Save.
   The new member definition details are displayed in the Members window.

4.5 Member Definition Dependencies

You can view the dependencies of Members.

To view the dependency of member in the Members window:

1. Select the checkbox adjacent to the Alphanumeric Code of the Member, whose dependency is to be viewed.
2. Click **Check Dependencies** button in the toolbar. The Check Dependencies button is disabled if you have selected multiple members.
   The Members Dependency Information window is displayed with the dependency details.

Copyright © 2011, 2026, Oracle and/or its affiliates.

4.6 Deleting Member Definition

You cannot delete predefined members or the members which are the Nodes for a hierarchy.

To delete a Member in the Members window.

1. Select the checkbox adjacent to the Alphanumeric Code(s) of the Members, whose details are to be removed.
2. Click **Delete** button in the Dimension Members tool bar.
3. Click **OK** in the information dialog to confirm deletion.

Copyright © 2011, 2026, Oracle and/or its affiliates.

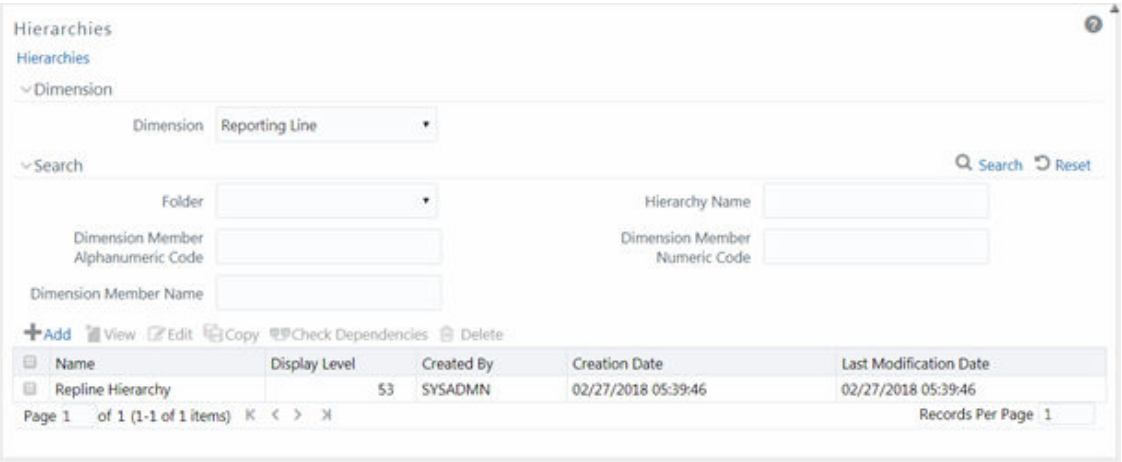
5

Hierarchy Maintenance

Hierarchies refer to dimension members that are arranged in levels, with each level representing the aggregated total of the data from the level below. One dimension type can have multiple hierarchies associated with it. Hierarchies are available within the Dimension Management section of Infrastructure system.

You can access Hierarchies window by expanding **Unified Analytical Metadata** and **Dimension Management** within the tree structure of LHS menu and selecting **Hierarchy Maintenance**.

Figure 5-1 Hierarchies window



The Hierarchies window displays the list of Hierarchies created in all public folders, shared folders to which you are mapped and private folders for which you are the owner, along with other details such as the Name, Display level, Created By, Creation Date, and Last Modification Date. For more information on how object access is restricted, see Object Security in AMHM module section.

You can also search for a specific Hierarchy definition based on Folder, Hierarchy Name, Dimension Member Alphanumeric Code, Dimension Member Numeric Code, or Dimension Member Name and view the existing definitions within the system.

- Adding Hierarchy Definition
  In the Hierarchies window, you can create Hierarchy Definition up to 15 levels by default. The maximum permissible levels are up to 58 Hierarchies. To create a hierarchy, the Write role should be mapped to your user group.
- Viewing Hierarchy Definition
  You can view individual Hierarchy Definition details at any given point.
- Modifying Hierarchy Definition
  You can modify the Name, Description, Folder, Access Type, Automatic inheritance, Display Signage, Show Member Code, Initial Display level, Orphan branch, Show hierarchy details in Edit – Hierarchy Details window.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Copying Hierarchy Definition
  The Copy Hierarchy Definition facilitates you to quickly create a new Hierarchy Definition based on the existing attributes or by updating the values of the required hierarchies.
- Hierarchy Definition Dependencies
  You can view the dependencies of Hierarchies.
- Deleting Hierarchy Definition
  You can remove the Hierarchy Definitions which are not required in the system by deleting from the Hierarchies window.

5.1 Adding Hierarchy Definition

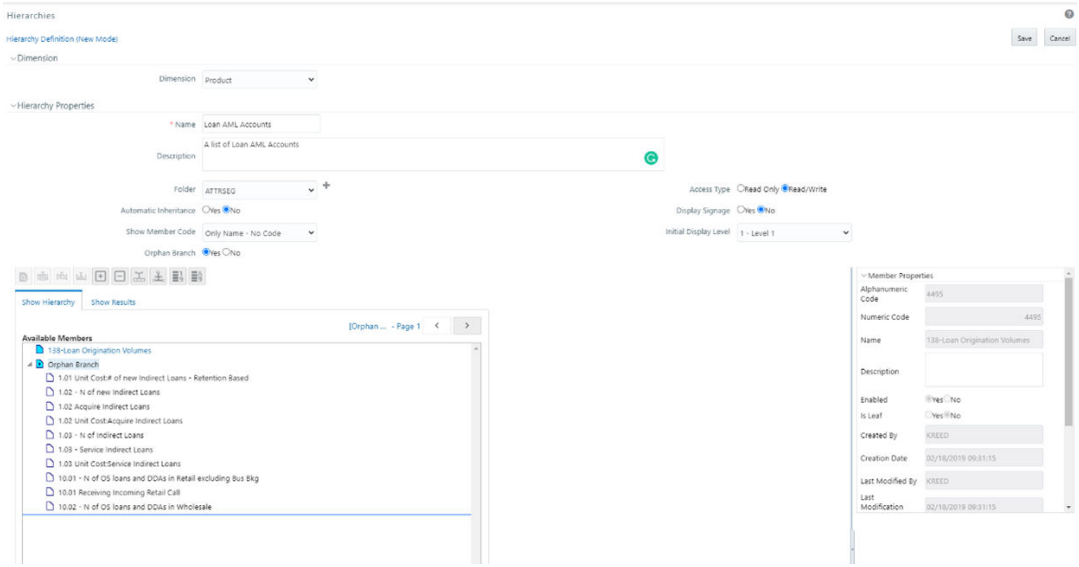
In the Hierarchies window, you can create Hierarchy Definition up to 15 levels by default. The maximum permissible levels are up to 58 Hierarchies. To create a hierarchy, the Write role should be mapped to your user group.

> **Note**
>
> When an AMHM Hierarchy is created, implicitly a UAM Business hierarchy also gets created and will be listed in the Summary window of Business Hierarchy.

To create a Hierarchy definition in the Hierarchies window:

1. Click the **Add** button in the Hierarchies toolbar.
   The New – Hierarchy Details window is displayed.

Figure 5-2 Add Hierarchies window



2. Select Dimension from the drop-down list.
   The selected Dimension from the New – Hierarchy Details window is displayed as the default dimension for which member has to be defined.

   The following table describes the fields in the Hierarchy Properties window.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Field	Description
Fields marked in red asterisk (*) are mandatory.	
Name	Enter the **Name** of the Hierarchy. **Note**: The characters &' " are restricted.
Description	Enter the required **Description** for the Hierarchy. **Note**: The characters ~&+' " @ are restricted.
Folder	Select the folder where the hierarchy is to be stored from the drop-down list. The Folder selector window behavior is explained in User Scope section. Click the **Add** button to create a new private folder. The Segment Maintenance window is displayed. For more information, see Segment Maintenance. **Note**: You can select **Segment/Folder Type** as Private and the **Owner Code** as your user code only.
Access Type	Select the Access Type as **Read Only** or **Read/ Write**. **Read-Only**: Select this option to give other users access to only view the hierarchy definition. **Note**: A user with Phantom and Write role can modify or delete the hierarchy even though the access type is selected as Read-only. **Read/Write**: Select this option to give all users the access to view, modify (including Access Type), and delete the hierarchy definition.
Automatic Inheritance	Click **Yes** to inherit the hierarchy properties of the Parent to the Child. Click **No** if you want to define a new hierarchy.
Display Signage	Click **Yes** to display the Signage to the right hand side of the member in the Show hierarchy panel. Else, click **No**.
Show Member Code	Select from the drop-down list as one of the following: • **Alphanumeric Code to Left of Name**: Displays Alphanumeric Code on the Left side of Member Name. • **Alphanumeric Code to Right of Name**: Displays Alphanumeric Code on the Right side of Member Name. • **Only Name - No Code**: Displays only the Member Name. • **Numeric Code to Left of Name**: Displays the Numeric Code on the Left side of Member Name. • **Numeric Code to Right of Name**: Displays the Numeric Code on the Right side of Member Name.
Initial Display Level	Select the Initial Display level from the drop-down list.
Orphan Branch	Click **Yes** to display the Orphan Branch in the Show Hierarchy panel. Otherwise, click **No**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

3. To add Child under the Show Hierarchy tab:
The TREE _NODE_LIMIT needs to be set in the AMHMConfig.properties file is moved to the Hierarchy database. There are two ways to render members of a hierarchy. One is the paginated mode, and the other is the non-paginated mode. The method is decided based on the number of nodes for a particular hierarchy.
   - If the number of members is more than TREE_NODE_LIMIT (configured in AMHM property tables in CONFIG schema and not in the properties file), members are loaded in paginated mode.
   - If the number of members is less than TREE_NODE_LIMIT (configured in AMHM property tables in CONFIG schema and not in the properties file), members are loaded non-paginated mode.
   - If TREE_NODE_LIMIT is not configured in AMHM property tables, then this value defaults to 5000. Any change in this table requires a server restart, and the values in the property file does not affect 8.1.0.0.0+ environments.

   Select the **Pagination** icon to view more options under the available components. Click a record to enable the Pagination buttons.
   a. Right-click in the Show Hierarchy tab.
   b. Select Add Child option and the Add Member window are displayed.
   c. Select the required Member and click **Move** button. The Member is displayed in the Selected Members panel. Click **Move All** button to select all Members which are shown in the Show Members pane. Click to select all nodes/members in the server. You can click **Remove** button to deselect a Member or click **Remove All** button to deselect all the Members.
   You can click the **Search** icon to search for the required member using Alphanumeric code, Numeric Code, Name, Description, Attribute Name, or Attribute Value.

      You can also click the **Toggle Numeric Code** icon to toggle the display of Numeric Code left, right, or name and click **Names Left** icon to display Alphanumeric Code left, right, or name.
   d. Click **OK**.
   The selected Member is displayed as Child under **Show Hierarchy** panel in the New – Hierarchy Details window.
4. To add Sibling:
   a. Right-click on the Child and select the option Add Sibling.
   The Add Member window is displayed.
   b. Select the required Member and click the **Move** button.
   The Member is displayed in the **Selected Members** panel. You can click **Move All** button to select all Members which are shown in the Show Members pane. Click **Select All** icon to select all nodes/ members in the server.
   c. You can click the **Remove** icon to deselect a Member or click **Remove All** icon to deselect all the Members. You can also Click **Search** icon to search for the required member.
   d. Click Apply.
   The selected Member is displayed as **Sibling** below the **Parent** under Show Hierarchy panel in the New – Hierarchy Details window.
5. To add Leaf under a Parent, Child, or Sibling:
   a. Right-click the Parent or Child and select **Add Leaf**.
   The Add Member window is displayed.
   b. Select the required Member and click the **Move** button.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The Member is displayed in the **Selected Members** panel. You can click the **Move All** button to select all Members which are shown in the Show Members pane. Click to select all nodes/ members in the server.

You can click the **Remove** button to deselect a Member or click **Remove All** button to deselect all the Members. You can also Click the **Search** icon to search for the required member.

c. Click **Apply**.
The selected Member is displayed as Leaf below the Parent or Sibling under **Show Hierarchy** panel in the New – Hierarchy Details window.

6. To define Level Properties:
   a. Select **Level Properties** from the options under Parent, Child, Sibling or Leaf and the Level Properties window is displayed.
   b. Enter the valid **Name** and **Description** in the respective fields.
   c. Click **OK** and the Levels defined are displayed in the drop-down in **Initial Level Display** field in **Hierarchy Properties** grid in New – Hierarchy Details window.
7. To cut and paste Child or Sibling:
   a. Right-click on any node and select **Cut**.
   b. Right-click on any node and **Paste as Child** or **Paste as Sibling**.
8. To Delete and Undelete:
   a. Right-click on the node to be deleted and select **Delete Node**.
   The node deleted is stroked out.
   b. Right-click and select **UnDelete** to cancel deletion of the node.
9. To add Child / Sibling / leaf:
   a. Right-click on any node and select **Create and add Child**.
   The New - Member Details window is displayed. For more information, see Add Member Definition.
   b. b. Right-click on any node and select **Create and add Sibling**.
   c. Right-click on any node and select **Create and add leaf**.
10. To view the Member Properties and Member Attributes of a node in the Show Hierarchy panel:
    a. Click the **Left Arrow** button and the Member Property grid is displayed.
    b. Click on a Member.
    The properties such as Alphanumeric code, Numeric Code, Name, Description, Enabled, Is Leaf, Created By, Creation Date, Last Modified By, Last Modification Date, Attribute, and Value of the selected Member are displayed in the Member Properties and Member Attributes grids.

    In the Hierarchies window you can also:

    - Click the **Collapse** icon to collapse the members under a node.
    - Click the **Expand** icon or **Branch** icon to expand a branch or collapse a branch.
    - Click the **Focus** icon or **Defocus** icon to focus or defocus a selected node except the root node.
    - Click **Names Right** icon or **Name Left** icon to view the name of members right or left.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Click the **Numeric Right** icon or **Numeric Left** icon to view the Numeric code values of members right or left.
- Click the **Show Member Code** icon or **Show Member Name** icon to show code or show name of the members.
- Click the **Advanced Properties** button to view the Advanced Properties of the nodes.

11. Click **Save** in the New – Hierarchy Details window to validate and capture the entries. The Audit Trail section at the bottom of the window displays the metadata about the Hierarchy with the option to add additional information as comments. The User Comments section facilitates you to add or update additional information as comments.

5.2 Viewing Hierarchy Definition

You can view individual Hierarchy Definition details at any given point.

To view the existing Hierarchy Definition details in the Hierarchies window:

1. Select the checkbox adjacent to the Hierarchy Name.
2. Click the **View** button in the Hierarchies tool bar. The View button is disabled if you have selected multiple Hierarchies.
   The View – Hierarchy Details window is displayed with all the Hierarchy details.

   In the View – Hierarchy Details window you can click the **Search** icon to search for a member using the Alphanumeric Code, Numeric Code, or Member Name in the Search dialog.

> **Note**
>
> The search functionality of this **Search** icon will not return any values if you search for a node in the Orphan Branch of the hierarchy.

5.3 Modifying Hierarchy Definition

You can modify the Name, Description, Folder, Access Type, Automatic inheritance, Display Signage, Show Member Code, Initial Display level, Orphan branch, Show hierarchy details in Edit – Hierarchy Details window.

> **Note**
>
> When you modify a Hierarchy, the implicitly created UAM Business Hierarchy will also get updated.

To modify the Hierarchy Definition:

1. Select the checkbox adjacent to the Hierarchy Name whose details are to be updated.
2. Click the **Edit** button in the Hierarchies tool bar.
   The **Edit** button is disabled if you have selected multiple Members. The Edit – Hierarchy Details window is displayed. In the Edit – Hierarchy Details window you can click the **Search** button to search for a member using the Alphanumeric Code, Numeric Code, or Member Name in the Search dialog. Edit the Hierarchy details as required.

Copyright © 2011, 2026, Oracle and/or its affiliates.

For more information, see Add Hierarchy Definition.

3. Click **Save** and save the changes.

5.4 Copying Hierarchy Definition

The Copy Hierarchy Definition facilitates you to quickly create a new Hierarchy Definition based on the existing attributes or by updating the values of the required hierarchies.

To copy an existing Hierarchy Definition in the Hierarchies window:

1. Select the checkbox adjacent to the Hierarchy name whose details are to be duplicated.
2. Click the **Copy** button in the Hierarchies toolbar to copy a selected Hierarchy definition.

   The **Copy** button is disabled if you have selected multiple Hierarchies. The Copy – Hierarchy Details window is displayed.
   In the Copy – Hierarchy Details window you can click the **Search** button to search for a member using the Alphanumeric Code, Numeric Code, or Member Name in the Search dialog.
3. In the Copy – Hierarchy Details window you can:
   - Create new hierarchy definition with existing variables. Specify a new **Hierarchy Name**. Click **Save**.
   - Create new hierarchy definition by updating the required variables. Specify a new Hierarchy Name and update the required details.
     For more information, see Add Hierarchy Definition.
4. Click **Save**.

   The new Hierarchy definition details are displayed in the Hierarchies window.

5.5 Hierarchy Definition Dependencies

You can view the dependencies of Hierarchies.

To view the dependency of Hierarchy in the Hierarchies window:

1. Select the checkbox adjacent to the Hierarchy Name.
2. Click the **Check Dependencies** button in the Hierarchies toolbar.

   The **Check Dependencies** button is disabled if you have selected Hierarchy definitions.

   The Hierarchies Dependency Information window is displayed.

5.6 Deleting Hierarchy Definition

You can remove the Hierarchy Definitions which are not required in the system by deleting from the Hierarchies window.

> **Note**
>
> When you delete an AMHM Hierarchy, the implicitly created UAM Business Hierarchy will also get deleted, if it is not used in higher objects.

To delete a Hierarchy Definition:

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. Select the checkbox adjacent to Hierarchy Name(s) whose details are to be removed.
2. Click the **Delete** button in the Hierarchies tool bar.
3. Click **OK** in the information dialog to confirm deletion.

To delete an existing Business Hierarchy in the Business Hierarchy window:

1. Select the checkbox adjacent to the required Business Hierarchy code.
2. Click the **Delete** button from the Business Hierarchy tool bar.
   A confirmation dialog is displayed.
3. Click **OK**. The Business Hierarchy details are marked for delete authorization.

Copyright © 2011, 2026, Oracle and/or its affiliates.

6
Filter

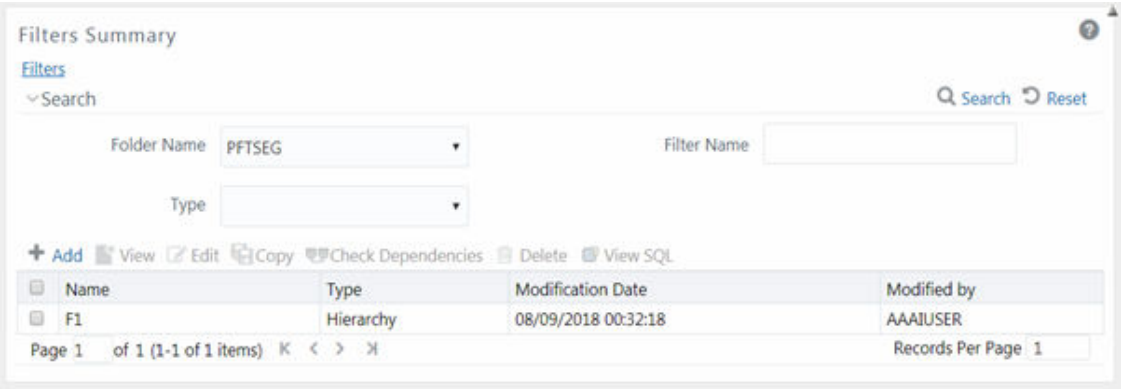
Filters in the Infrastructure system allow you to filter metadata using the defined expressions.

You can access Filters by expanding **United Analytical Metadata** section within the tree structure of LHS menu and selecting **Filter**.

Based on the role that you are mapped to, you can access read, modify or authorize Filters window. For all the roles and descriptions, see Appendix A. The roles mapped to Filters are as follows:

- Filter Access
- Filter Advanced
- Filter Authorize
- Filter Phantom
- Filter Read Only
- Filter Write

Figure 6-1 Filter Summary window



The Filters Summary window displays the list of Filters created in all public folders, shared folders to which you are mapped and private folders for which you are the owner, along with the other details such as the Name, Type, Modification Date, and Modified By.

For more information on how object access is restricted, see Object Security in Dimension Management module section.

You can also search for a specific Filter definition based on Folder Name, Filter Name, or Type and view the list of existing definitions within the system. If you have selected Hierarchy from the Type drop-down list, the Dimension drop-down list is also displayed.

- Adding Filter Definition
  This option allows you to add a filter. Filter can be of 4 types namely, Data Element, Hierarchy, Group, and Attribute. To create filter definition, the Write role should be mapped to your user group.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Viewing Filter Definition
  You can view individual Filter details at any given point.
- Modifying Filter Definition
  This option allows you to modify the details of Filters.
- Copying Filter Definition
  The Copy Filter Definition facilitates you to quickly create a new Filter Definition based on the existing parameters or by updating the values.
- Checking Dependencies
  You can view the dependencies of a defined Filter. You can use filter in a Run definition. However, the Run definitions are not shown as dependent objects when you check dependency for a filter. This is a limitation.
- Viewing SQL of Filter
  You can view the corresponding SQL of a defined filter.
- Deleting Filter Definition
  You can remove the Filter Definitions which are not required in the system by deleting from the Filters Summary window.

6.1 Adding Filter Definition

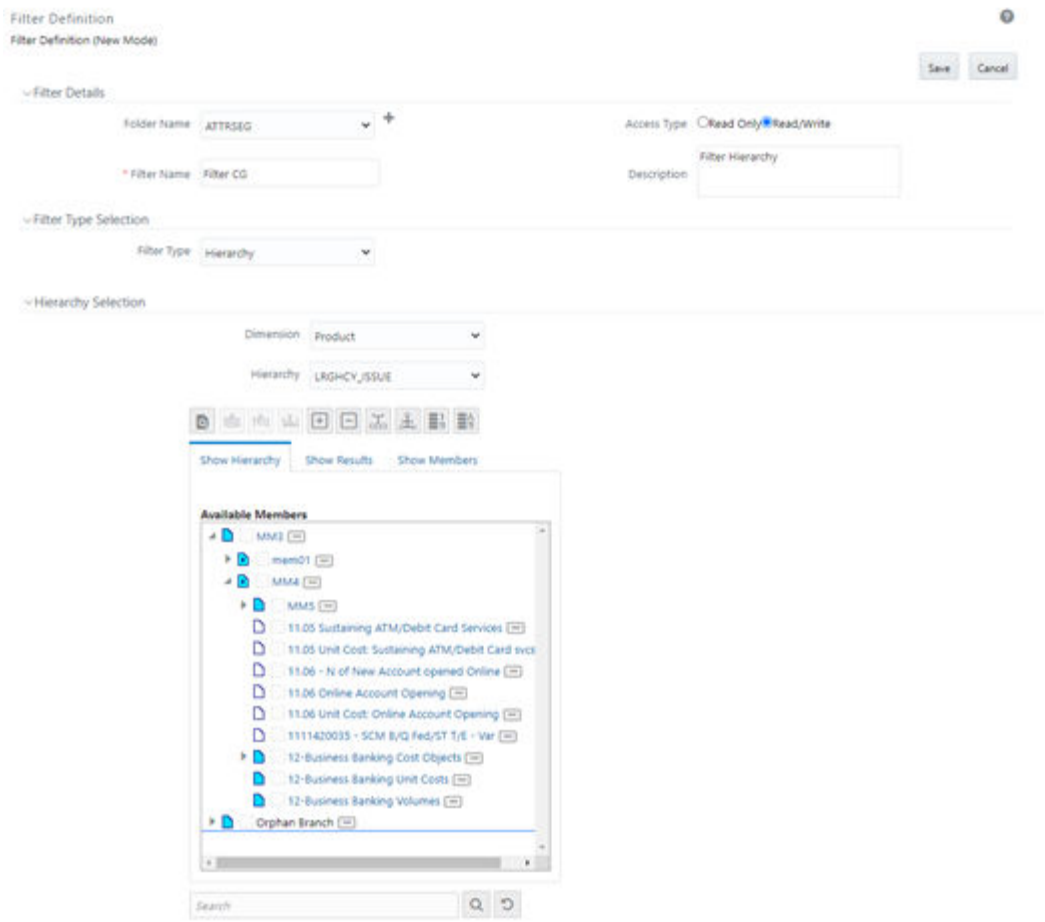
This option allows you to add a filter. Filter can be of 4 types namely, Data Element, Hierarchy, Group, and Attribute. To create filter definition, the Write role should be mapped to your user group.

To create a new filter from the Filters Summary window:

1. Click the **Add** button in the Filters toolbar.
   The Filter Definition window is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 6-2 Filter Definition New window



2. Enter the Filter Details section details as tabulated:
The following table describes the fields in the Filter Definition window.

Table 6-1 Fields in the Filter Definition window and their Description

Field	Description
Filter Details	
Folder Name	Select the Folder Name where the Filter is to be stored from the drop-down list. The Folder selector window behavior is explained in User Scope section. Click to create a new private folder. The Segment Maintenance window is displayed. For more information, see Segment Maintenance section. **Note**: You can select **Segment/Folder Type** as Private and the **Owner Code** as your user code only.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 6-1 (Cont.) Fields in the Filter Definition window and their Description

Field	Description
Access Type	Select the Access Type as **Read Only** or **Read/ Write**. **Read-Only**: Select this option to give other users the access to only view the filter definition. **Note**: A user with Phantom and Write role can modify or delete the filter even though the access type is selected as Read-only. **Read/Write**: Select this option to give all users the access to view, modify (including Access Type) and delete the filter definition.
Filter Name	Enter the filter name in the **Filter Name** field. Note: The characters &' " are restricted.
Description	Enter the description of the filter in the **Description** field. Note: The characters ~&+' " @ are restricted.

3. From the Filter Type Selection pane, select the **Filter Type** from the drop-down list. There are four different Filter Types available in the Filter Type Selection grid as tabulated. Click the links to navigate to the appropriate sections.

   The following table describes the fields in the Filter Type pane.

Table 6-2 Fields in the Filter Type pane and their Descriptio

Filter	Description
Data Element	**Data Element Filter** is a stored rule that expresses a set of constraints. Only columns that match the data type of your Data Element selection are offered in the Data Element drop-down list box. **Example**: Balances between 10,000 and 20,000 Accounts opened in the current month Loans with amortization terms greater than 20 years. Data Element Filters can access most instrument columns and most columns in the Management Ledger. Data Element Filters are used within other OFSAA rule types (e.g., Allocation rules, Transfer Pricing rules, Asset | Liability Management rules, and others).
Hierarchy	**Hierarchy Filter** allows you to utilize rollup nodes within a Hierarchy to help you exclude (filter out) or include data within an OFSAA rule. **Example**: You might want to process data for a specific set of divisions or lines of business where you have a Hierarchy rule that expresses those divisions or lines of business as rollup nodes. A Hierarchy Filter could be constructed to "enable" the Commercial and Retail lines of business while NOT enabling the Wealth Management line of business. Each of these lines of business might include a handful or even thousands of cost centers. When incorporated into an OFSAA processing rule, this Hierarchy Filter would include every cost center in the Commercial and Retail lines of business.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 6-2 (Cont.) Fields in the Filter Type pane and their Descriptio

Filter	Description
Group	**Group Filters** can be used to combine multiple Data Element Filters with a logical "AND". **Example**: If Data Element Filter #1 filtered on mortgage balances greater than 100,000 and Data Element Filter #2 filtered on current mortgage interest rates greater than 6%, you could construct a Group Filter to utilize both Data Filters. In this case, the resulting Group Filter would constrain your data selection to mortgage balances greater than 100,000 AND current mortgage interest.
Attribute	**Attribute Filters** are created using defined Attributes. Attribute filters facilitates you to filter on one or more Dimension Type Attributes. For each attribute, you can select one or more values. **Example**: Consider a filter that selects all records where the dimension Common Chart of Account member represents an attribute value Expense account, i.e., the attribute **"Account Type" = Expense**. Now, using Attribute Filters, you can specify complex criteria as given below: Common Chart of Accounts where the Account Type attribute is Earning Assets or Interest-bearing Liabilities, and the Accrual Basis attribute is Actual/Actual Also, You could further refine the filter by adding another condition for: Organizational Unit where the Offset Org ID is a specific Org member The Filter then saves these criteria rather than the member codes which meet the criteria at the time the Filter is saved. During execution, the engine dynamically selects all records from your processing table (e.g. Mortgages, Ledger, etc.), which meet the specified member attribute criteria.

4. Click **Save** after the required filter conditions are defined to save the Filter definition.

- Define Data Element Filter
  When you have selected the Filter Type as Data Element, define the Filter conditions by doing the following in the Data Element Selection section:
- Define Hierarchy Filter
  When you have selected the Filter Type as Hierarchy, define the Filter conditions by doing the following in the Hierarchy Selection section:
- Define Group Filter
  When you have selected the Filter Type as Group, define the Filter conditions by doing the following in the Data Element Filters grid:
- Define Attribute Filter
  When you have selected the Filter Type as Attribute, define the Filter conditions by doing the following in the Attribute Selection section:

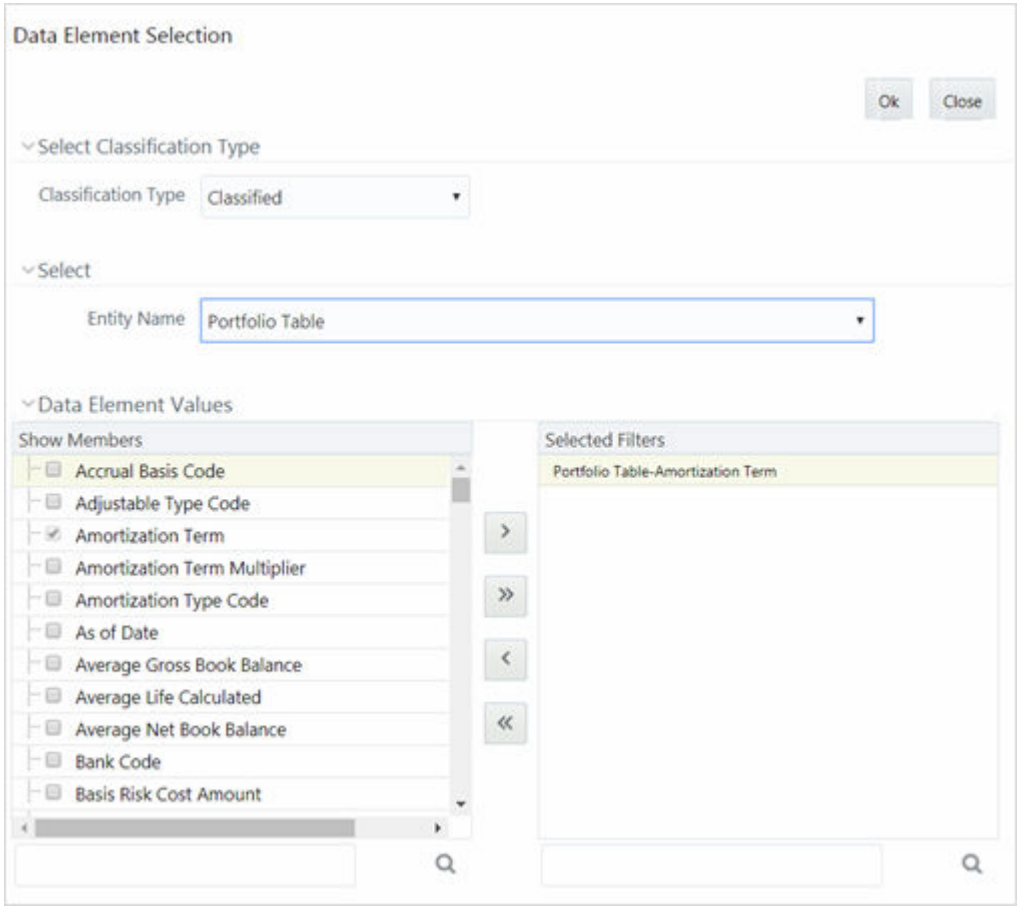
6.1.1 Define Data Element Filter

When you have selected the Filter Type as Data Element, define the Filter conditions by doing the following in the Data Element Selection section:

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. In the Data Element Selection section, click the **View Attributes** icon.
The Data Element Selection window is displayed.

Figure 6-3 Data Element Selection window



- Select any of the following Filter **Classification Type** from the drop-down list:
  - **Classified**: This is the default selection and displays all the classified EPM specific entities. If you are an EPM user, you need to select this option while defining Data Element Filter to list all the related entities.
  - **Unclassified**: This option displays all the non-classified i.e. non EPM specific entities. If you are a non EPM user, you need to select this option while defining Data Element Filter to list all the related entities.
  - **All**: This option will select all the tables available in the selected Information Domain irrespective of whether an entity has its table is classified or not.
- Select the required database table from the **Entity Name** drop-down list. The associated members are displayed in the Show Members section.
- Select the required member and click the **Move** button. The member is listed in the Selected Members panel. Click the **Move All** button to move all Members.
You can click **Remove** button to deselect a Member or click **Remove All** button to deselect all Members.
- Click **OK**. The selected Data Elements are displayed in the **Data Element Selection** field.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Select the **Filter Method** from the drop-down list.
For each column you wish to include in your Data Filter definition, you must specify one of the following Filter Method:

The following table describes the fields in the Data Filter Definition.

Table 6-3 Fields in the Data Filter Definition window and their Description

Filter	Description
Specific Values	**Specific Values** are used to match a selected database column to a specific value or values that you provide. You may either include or exclude Specific Values. You can add additional values by clicking the **Add** button. Click adjacent to **Add** button to add 3, 5, 10 rows by selecting the checkbox adjacent to 3, 5, or 10 respectively. You can add custom number of rows by specifying the number in the text box provided, as shown and click the **Select** button. **Figure 6-4 Specific Values selected**  To remove a row, select the checkbox and click **Delete**. When comparing Specific Values for a character type column, you must provide Specific Values that are character strings. When comparing Specific Values for a date type column, you must provide Specific Values that are dates (the application displays a Calendar control). When comparing Specific Values for a numeric column, you must provide Specific Values that are numbers. Select **Include Values** or **Exclude Values** to include or exclude the selected values.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 6-3 (Cont.) Fields in the Data Filter Definition window and their Description

Filter	Description
Ranges	**Ranges** are used to match a selected database column to a range of values or to ranges of values that you provide. You may either include or exclude Range values. Range Type is available for OFSA Datatype Term, Frequency, Leaf, Code, and Identity and Column Datatype Date, Numeric and Varchar. You can add additional values by clicking the **Add** button. Click adjacent to **Add** button to add 3, 5, 10 rows by selecting the checkbox adjacent to 3, 5, or 10 respectively. You can add custom number of rows by specifying the number in the text box provided, as shown and click . To remove a row, select the checkbox and click **Delete**. If the Column Datatype is VARCHAR, provide Specific Values (alphanumeric) that are character strings. If the Column Datatype is DATE, provide Specific Values that are dates (the application displays a Calendar control). If the Column Datatype is Numeric, provide Specific Values that are numbers. If OFSA Datatype is LEAF, provide either numeric values or click to select the numeric member ids. If OFSA Datatype is CODE, provide either numeric values or click to select the numeric member ids. If OFSA Datatype is IDENTITY, provide specific numeric values. However, no validation is done during save to validate the input value for a valid identity code. Select **Include Values** or **Exclude Values** to include or exclude the selected values
Another Data Element	**Another Data Element** is used to match a selected database column to another database column. When constructing an Another Data Element Filter Method, you may only compare a column to other columns that you have already selected (the Data Element drop-down list box will only contain columns that you have already selected). You may use any of the following operators when choosing the Another Data Element Filter Method: =, <> (meaning "not equal to"), <, >, <=, or >=.
Expression	**Expression** is used to match a selected database column to the results of an OFSAAI Expression rule. You may any of the following operators when choosing the Expression Filter Method: =, <> (meaning "not equal to"), <, >, <=, or >=.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Click **Add** to list the completed filter conditions in the Filter Conditions grid.
- Click **Update** after modifying a filter condition to update in the Filter Conditions grid.
- Click the **Sort Ascending** or **Sort Descending** icons to move a selected Filter Condition up or down.
- Click the **Delete** button to delete selected individual Filter Conditions records.

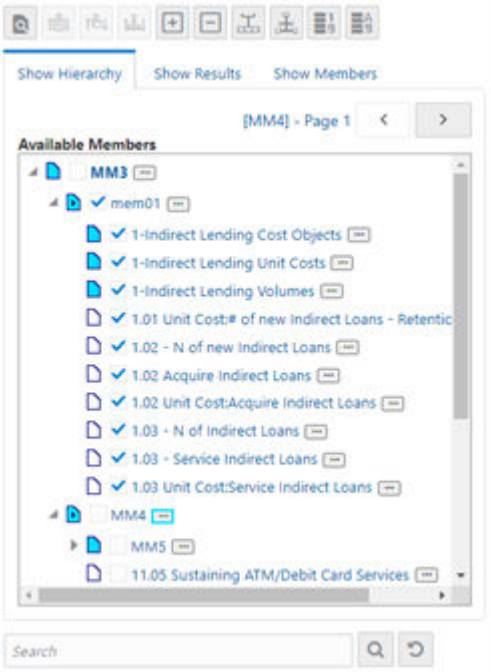
3. Click **Add** or **Edit** in the Filter Definition window if you are creating a new or updating an existing Filter definition.
4. Click **Save** to validate the entries and save the filter details.

6.1.2 Define Hierarchy Filter

When you have selected the Filter Type as Hierarchy, define the Filter conditions by doing the following in the Hierarchy Selection section:

1. Select the required **Dimension** from the drop-down list.
2. Select the associated **Hierarchy** from the drop-down list. You can select More to search for a specific Hierarchy in the Hierarchy more dialog.
3. Select any combination of rollup points and leaf (last descendent child) values.

Figure 6-5 Show Hierarchy Tab



The Show Hierarchy tab displays the leaves in each node in ascending order of Members.

In order to sort the nodes alphabetically, `HIERARCHY_IN_FILTER_SORT-$INFODOM$-$DIMENSION_ID$=$VALUE$` in the `AMHMConfig.properties` file present in the deployed location should be set as Y. You should add such entry for all the required Dimension IDs for the sort functionality to work for those dimensions.

For example: `HIERARCHY_IN_FILTER_SORT-OFSAAINFO-4345=Y`

Copyright © 2011, 2026, Oracle and/or its affiliates.

Restart servers after making any change in AMHMConfig.properties file for the change to take effect.

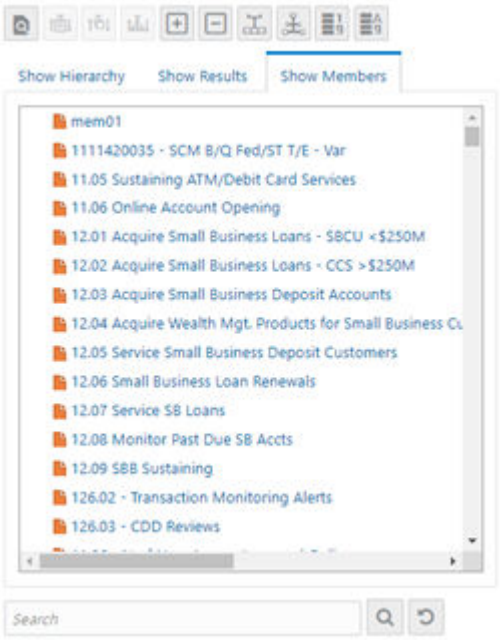
> **Note**
>
> Select the **Pagination** icon to view more options under the available components. Click the **More Options** (three dots) icon to enable the Pagination buttons.

From this pane, you can:

- Click **Search** button to search for a hierarchy member using Dimension Member Alphanumeric Code, Dimension Member Numeric Code, Dimension Member Name, or Attribute and by keying in Matching Values in the Search dialog. The search results are also displayed in the ascending order of Member Names.
- Click **Collapse** icon to collapse the members under a node.
- Click **Name Right** or **Name Left** icons to view the name of members right or left.
- Click **Numbers Right** or **Numbers Left** icons to view the Numeric code values of members right or left.
- Click **Show Member Code** or **Show Member Name** icons to show code or show name of the members.
- Click **Focus** or **Defocus** icons to focus or defocus a selected node except the root node.
  You can also click the **Search** button to find a member present in the nodes list using key words. For large tree (nodes>5000), this search will not return any value if the tree is not expanded.

4. Click **Save** to validate the entries and save the filter details.

Figure 6-6 Show Members Tab



Copyright © 2011, 2026, Oracle and/or its affiliates.

The **Show Members** tab displays all the selected nodes in a list view, which helps you visualize all the selected nodes as a list rather than as a tree. Currently, this feature is available in the **Edit** and **View** mode of the Hierarchy Filter.

6.1.3 Define Group Filter

When you have selected the Filter Type as Group, define the Filter conditions by doing the following in the Data Element Filters grid:

1. Select the checkbox(s) adjacent to the required member names in the Available Filters section and click the **Move** button. The selected members are displayed in the Selected Filters section. Click **Move All** button to select all the Members.
   You can click **Remove** button to deselect a Member or click **Remove All** button to deselect all the Members.

   You can also click the **Search** button to search for a member in the Data Element Filter Search dialog using Folder Name and Filter Name.
2. Click **Save** to validate the entries and save the filter details.

6.1.4 Define Attribute Filter

When you have selected the Filter Type as Attribute, define the Filter conditions by doing the following in the Attribute Selection section:

1. Select the required Dimension from the drop-down list.
2. Select the associated Attribute from the drop-down list. The list consists of only Dimension Type attributes for selection.
3. Click the **Show Attribute** icon in the Attribute Values grid.

   The Attribute Values window is displayed.

   In the Attribute Values window, the **Dimension** field under Dimension grid is auto populated with the Dimension name with which the selected Attribute is defined and is non-editable. In the Search grid you can search for Attribute Values depending on Alphanumeric Code, Numeric Code, or Name.
4. Select the checkbox(s) adjacent to the Alphanumeric Codes of the required Attribute Values in the Attribute Values grid and click **OK**. The Attribute Values grid displays the selected attribute values.

   Select Attribute Value(s) in the Attribute Values grid and click the **Delete** icon to delete it.

   You can use the Attribute Values present in the Attribute Values grid to generate conditions.
5. Click the **Add** button in the Attribute Values grid. The Filter Conditions grid is populated with the filter condition using all the Attribute values.

   You cannot define two conditions using the same attributes. Because conditions are joined with a logical ‘AND’ and this will make the query invalid.

   In the Filter Conditions grid, you can select a condition to view the Attribute Values used to generate it and can update the condition.

   You can also click **View SQL** button to view the SQL statement in View SQL window. Click the **View Attributes** button to view a long filter condition in View Condition dialog.
6. Click **Save**. The Attribute Filter definition is saved.

Copyright © 2011, 2026, Oracle and/or its affiliates.

6.2 Viewing Filter Definition

You can view individual Filter details at any given point.

To view the existing Filter Definition details in the Filters Summary window:

1. Select the checkbox adjacent to the Filter Name.
2. Click the **View** button in the Filter tool bar.

   The View – Filter Details window is displayed with the filter details.

6.3 Modifying Filter Definition

This option allows you to modify the details of Filters.

To modify the Filter Definition:

1. Select the checkbox adjacent to the Filter Name whose details are to be updated.
2. Click the **Edit** button and the Edit – Filter Details window is displayed. Modify the required changes.

   For more information, see Add Filter Definition.
3. Click **Save** to save the changes.

6.4 Copying Filter Definition

The Copy Filter Definition facilitates you to quickly create a new Filter Definition based on the existing parameters or by updating the values.

To copy an existing Filter Definition in the Filters window:

1. Select the checkbox adjacent to the Filter Name which you want to create a copy.
2. Click the **Copy** button in the Filters tool bar. The **Copy** button is disabled if you have selected multiple checkboxes.

   The Copy – Filter Details window is displayed.
3. In the Copy – Filter Details window you can:
   - Create new filter definition with existing variables. Specify a new **Filter Name** and click **Save**.
   - Create new filter definition by updating the required variables. Specify a new Filter Name and update the required details. For more information, see Add Filter Definition. Click **Save**.

   The new filter definition details are displayed in the Filters Summary window.

6.5 Checking Dependencies

You can view the dependencies of a defined Filter. You can use filter in a Run definition. However, the Run definitions are not shown as dependent objects when you check dependency for a filter. This is a limitation.

To check the dependencies of a filter from the Filters Summary window:

1. Select the checkbox adjacent to the Filter Name.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Click the **Check Dependencies** icon in the Filters tool bar. The **Check Dependencies** button is disabled if you have selected multiple members.

   The Dependent Objects window is displayed with Object ID, Name, and ID Type of the dependent Objects.

6.6 Viewing SQL of Filter

You can view the corresponding SQL of a defined filter.

To view the SQL of a filter from the Filters Summary window:

1. Select the checkbox adjacent to the filter to view the SQL.
2. Click the **View SQL** button.

   The SQL equivalent of the selected filter is displayed in the View SQL window.

6.7 Deleting Filter Definition

You can remove the Filter Definitions which are not required in the system by deleting from the Filters Summary window.

> **Note**
>
> A filter definition with dependency cannot be deleted. However, if the dependent object is a Run Definition, you are able to delete the filter definition. This is a limitation.

To delete a Filter Definition:

1. Select the checkbox adjacent to the Filter Name whose details are to be removed.
2. Click the **Delete** button in the Filters tool bar.
3. Click **OK** in the information dialog to confirm deletion.

Copyright © 2011, 2026, Oracle and/or its affiliates.

7 Expression

An Expression is a user-defined tool that supplements other IDs and enables to manipulate data flexibly. Expression has three different uses:

- To specify a calculated column that the Oracle Financial Services Analytical Application derivatives from other columns in the database.
- To calculate assignments in data correction.
- To create calculated conditions in data and relationship filters.

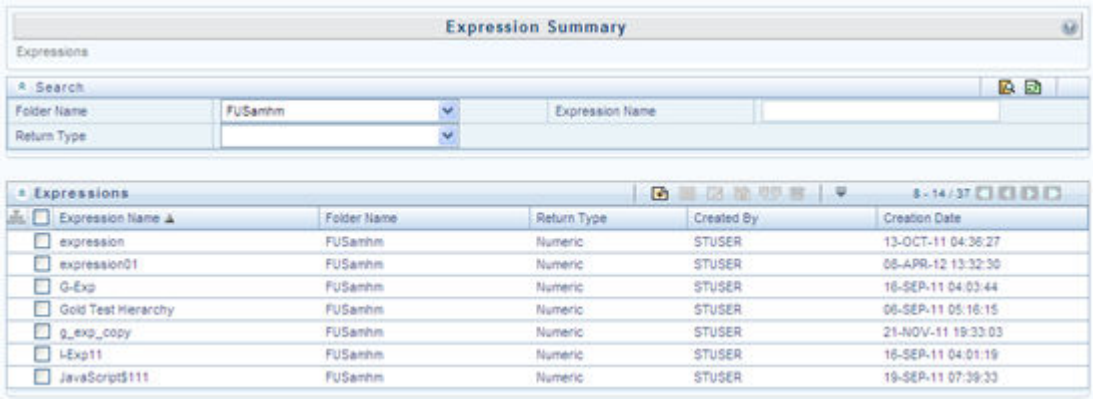
Example:- Calculations like average daily balances, current net book balance, average current net book balance, and weighted average current net rate can be created through Expressions.

Based on the role that you are mapped to, you can access read, modify or authorize Expression window. For all the roles and descriptions, see Appendix A.

The roles mapped to Expression are as follows:

- Expression Access
- Expression Advanced
- Expression Authorize
- Expression Phantom
- Expression Read Only
- Expression Write

Figure 7-1 Expression Summary window



The Expression Summary window displays the list of pre-defined Expressions with other details such as the Expression Name, Folder Name, Return Type, Created By, and Creation Date. For more information on how object access is restricted, see Object Security in Dimension Management module section.

You can also search for a specific Expression definition based on Folder Name, Expression Name, or Return Type and view the list of existing definitions within the system.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Adding Expression Definition
  This option allows you to add an expression definition using variables, functions, and operators. The Write role should be mapped to your user group.
- Viewing Expression
  You can view individual Expression details at any given point.
- Modifying Expression
  You can modify the Expression details as required in the Edit – Expression screen.
- Copying Expression
  The Copy Expression facilitates you to quickly create a new Expression based on the existing parameters or by updating the values.
- Checking Dependencies
  You can view the dependencies of a defined Expression in the Expression Summary screen:
- Deleting Expression
  You can delete an expression which has Read/Write Access Type.

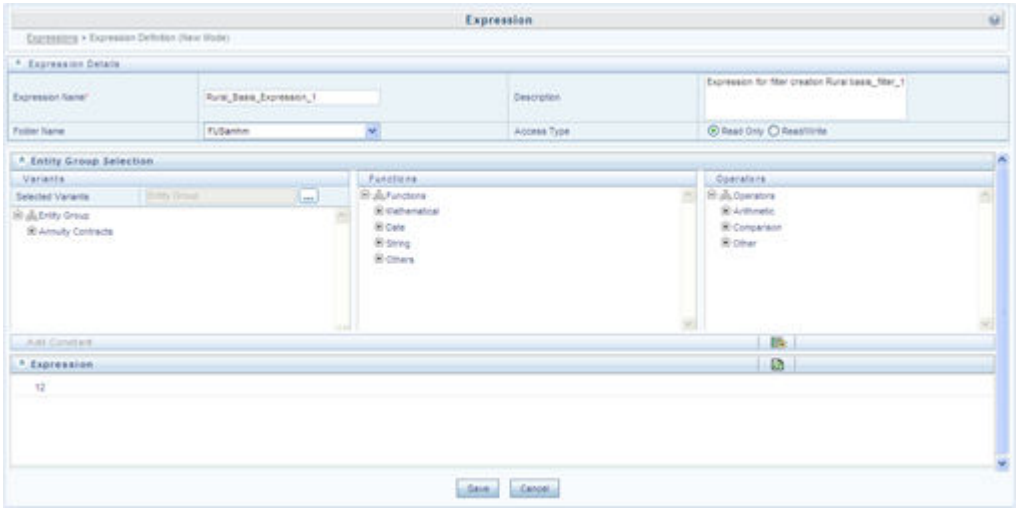
7.1 Adding Expression Definition

This option allows you to add an expression definition using variables, functions, and operators. The Write role should be mapped to your user group.

To create a new Expression from the Expressions Summary window:

1. Click the **Add** button in the Expressions Toolbar.
   The New - Expression window is displayed.

Figure 7-2 Expression Summary New window



2. In the Expression Details grid:
   - Enter the **Expression Name** and the required **Description**.

> **Note**
>
> Expression Name: The characters &' " are restricted in the name field.
> Description: The characters ~&+' "@ are restricted in the description field.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Select the **Folder Name** from the drop-down list.
  - The Folder selector window behavior is explained in User Scope section.
  - Click the **Add** icon to create a new private folder.
    The Segment Maintenance window is displayed. For more information, see Segment Maintenance.

> **Note**
>
> You can select **Segment/Folder Type** as Private and the **Owner Code** as your user code only.

- Select the **Access Type** as Read Only or Read/Write.
  - **Read-Only**: Select this option to give other users the access to only view the expression.

> **Note**
>
> A user with Phantom and Write role can modify or delete the expression even though the access type is selected as Read-only.

  - **Read/Write**: Select this option to give all users the access to view, modify (including Access Type) and delete the expression.

3. In the Entity Group Selection grid:
   - In the Variants section, click the **View** icon. The Variant Selection window is displayed.
     - Select the **Entity Type** and **Entity Name** from the drop-down lists.
     - Select the required member and click the **Move** icon.
       The member is displayed Selected Members list. Click the **Move All** icon to select all the Members.

       You can also click the **Remove** icon to deselect a Member or click **Remove All** icon to deselect all Members.
     - Click **OK**.
       The selected Entity Name and Members are displayed in the Variants section in the New Expression window.
   - In the Variant's section, click "+" to expand Entity Group and double-click to select the required Entity.
     The selected Entity is displayed in the Expression grid.
   - In the Function section, click "+" to expand Functions and select a function such as Mathematical, Date, String, or Others options.
     The selected Function is displayed in the Expression grid. For more information see Function Types and Functions.
   - In the Operators section, click "+" to expand Operators and select an operator such as Arithmetic, Comparison, or Others.
     The selected Operator is displayed in the Expression grid. For more information see Operator Types.
     - You can click **Add Constant** icon from the Add Constant grid to specify a Constant Value. Enter the numerical value and click .
   - In the Expression grid, you can right-click on the expression and do the following:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Click **Replace Expression** to replace the expression with a new one.
- Click **Insert Expression After** to insert a new expression after the selected expression.
- Click **Delete** to delete a selected expression.
- You can also click the **Clear Expression** icon in the Expression grid to clear the Expression.

4. Click **Save** to validate the entries and save the new Expression.

7.2 Viewing Expression

You can view individual Expression details at any given point.

To view the existing Expression details the Expression Summary window:

1. Select the checkbox adjacent to the Expression Name.
2. Click the **View** button in the Expressions tool bar.

   The View Expression window is displayed with the Expression details.

7.3 Modifying Expression

You can modify the Expression details as required in the Edit – Expression screen.

To modify the Expression:

1. Select the checkbox adjacent to the Expression Name whose details are to be updated.
2. Click the **Edit** button and the Edit – Expression window is displayed. Modify the required changes.

   For more information, see Add Expression Definition.
3. Click **Save** and upload the changes.

7.4 Copying Expression

The Copy Expression facilitates you to quickly create a new Expression based on the existing parameters or by updating the values.

To copy an existing Expression in the Expression Summary window:

1. Select the checkbox adjacent to the Expression Name which you want to create a copy.
2. Click the **Copy** button in the Expressions tool bar. The **Copy** button is disabled if you have selected multiple checkboxes.

   The Copy – Expression window is displayed.
3. In the Copy – Expression window you can:
   - Create new Expression with existing variables. Specify a new **Filter Name** and click **Save**.
   - Create new Expression by updating the required variables. Specify a new Expression Name and update the required details.
     For more information, see Add Expression Definition. Click **Save**.

     The new Expression details are displayed in the Expression Summary window.

Copyright © 2011, 2026, Oracle and/or its affiliates.

7.5 Checking Dependencies

You can view the dependencies of a defined Expression in the Expression Summary screen:

1. Select the checkbox adjacent to the required Expression Name.
2. Click the **Check Dependencies** button in the Expressions tool bar. The **Check Dependencies** button is disabled if you have selected multiple expressions.

   The Dependent Objects window is displayed with Object id, Name, and id type of the dependent Objects.

7.6 Deleting Expression

You can delete an expression which has Read/Write Access Type.

To delete an expression from the Expression Summary window:

1. Select the checkbox adjacent to the Expression Name(s) whose details are to be removed.
2. Click the **Delete** icon in the Expressions tool bar.
3. Click **OK** in the information dialog to confirm deletion.

Copyright © 2011, 2026, Oracle and/or its affiliates.

8

Simplified Batches

Simplified Batches provide a streamlined version of the functionality available in the broader standard Oracle Financial Services Analytical Applications (OFSAA) batch module.

The Simplified Batch feature is designed to facilitate the creation, maintenance, and execution of batches for the EPM application engines and various other OFSAA components. You can use its drag-and-drop approach to task selection especially helpful in the management of large batches.

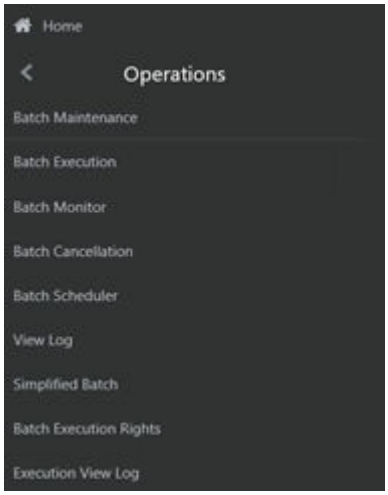
- Standard OFSAA Infrastructure Batching Functionality
  You can view, execute, schedule, and monitor Simplified Batches within the OFSAA Operations batch framework. To view the Simplified Batch summary page, navigate to **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**.
- Simplified Batch Details
  When you Add, Edit, or View a batch, the Simplified Batch details window is displayed. The Simplified Batch details window presents the following panes for defining and maintaining your batch:
- Creating a Simplified Batch
  To create a Simplified Batch Rule, follow these steps:
- Running a Simplified Batch
  To run a Simplified Batch from the Simplified Batch summary page, follow these steps:
- Viewing the Task Logs
  To view the processing log for each task in the batch, navigate to the log details from the Simplified Batch details page:
- Additional Batch Options

8.1 Standard OFSAA Infrastructure Batching Functionality

You can view, execute, schedule, and monitor Simplified Batches within the OFSAA Operations batch framework. To view the Simplified Batch summary page, navigate to **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 8-1 Accessing the Simplified Batch Window



- Batch Maintenance
  The **Batch Maintenance** window is used to view the Simplified Batches, but you cannot edit them. Create and modify them within the **Simplified Batch** window.
- Simplified Batches
  To view the **Simplified Batch** summary page, navigate to **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**. The Simplified Batch summary page displays a list of existing Simplified Batch rules. Using search criteria, you can control the set of batches that are displayed.

8.1.1 Batch Maintenance

The **Batch Maintenance** window is used to view the Simplified Batches, but you cannot edit them. Create and modify them within the **Simplified Batch** window.

Batch Execution

You can execute the Simplified Batches from the **Simplified Batch** summary page or the OFSAA Infrastructure UI by selecting **Operations**, and then **Batch Execution**.

Other Operations Batch Functionality

Other Operations batch functionality including Batch Scheduler, Batch Monitor, Batch Processing Report, Batch Cancellation, and View Log apply equally to Simplified Batches and to batches that have been built in the Batch Maintenance window.

For more information, see the OFS Analytical Applications Infrastructure User Guide.

Topics:

- Simplified Batches
- Simplified Batch details
- Creating a Simplified Batch
- Running a Simplified Batch
- Viewing the Task Logs
- Additional Batch Options

Copyright © 2011, 2026, Oracle and/or its affiliates.

8.1.2 Simplified Batches

To view the **Simplified Batch** summary page, navigate to **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**. The Simplified Batch summary page displays a list of existing Simplified Batch rules. Using search criteria, you can control the set of batches that are displayed.

When you Add, Edit, or View a rule, a **Simplified Batch** details window is displayed.

The Simplified Batch summary page has a Search pane and a summary table. When you first navigate to the Simplified Batch summary page, the batches stored within your preferred folder are presented on the summary page. This preferred folder is determined by the set of Application Preferences that were last saved in which you selected the Active for Master Maintenance check box. For more information, see the Application Preferences section.

The summary table presents all the batches that meet the search criteria. The Simplified Batch summary page offers several icons that allow you to create a new batch or perform different functions when a batch is selected.

Click the column header (the column name) to sort on a column in the Simplified Batch summary page.

Figure 8-2 Simplified Batch Summary Page



- Searching for a Simplified Batch
  The Search page allows you to restrict the batches visible within the summary table by any combination of Name, Folder, or Status. Search by Name is a wild card character search, that is, searching for Name-like pricing will find all batches that include pricing within their names.

8.1.2.1 Searching for a Simplified Batch

The Search page allows you to restrict the batches visible within the summary table by any combination of Name, Folder, or Status. Search by Name is a wild card character search, that is, searching for Name-like pricing will find all batches that include pricing within their names.

To search for a Simplified Batch Rule, follow these steps:

1. Enter the desired search criteria and click **Search**.
   Click Reset to clear any search criteria and refresh the Simplified Batch summary page.
2. The search results are displayed in a tabular format containing all of the Simplified Batch Rule curves that meet the search criteria. The Simplified Batch Rules summary page offers several icons to perform the different functions when a Simplified Batch Rule is selected.
   - **Add**: Click the **Add** icon to begin the process of building a new Simplified Batch. The Add icon is disabled if any rows in the summary table are selected.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **View**: Select a single row in the summary table to enable the View icon. Click View to view the detailed definition of a Simplified Batch on a read-only basis. The View icon is only enabled when a single batch is selected.
- **Edit**: Select a single row in the summary table to enable the Edit icon. Click Edit to modify an existing Simplified Batch. The Edit icon is only enabled when a single batch is selected.
- **Copy**: Select a single row in the summary table to enable the Copy icon. Click Copy to create a copy of an existing Simplified Batch. The Copy icon is only enabled when a single Simplified Batch is selected. You can also duplicate a batch using the Save As from the Simplified Batch Detail page.
- **Delete**: Select one or more rows in the summary table to enable the Delete icon. Click Delete to deletes the batch or batches are selected.
- **Run**: Select a single row in the summary table to enable the Run icon. Click Run to execute the selected Simplified Batch rule. The Run icon is only enabled when a single Simplified Batch rule is selected. The As-of-Date and other parameters applicable to each task within a Simplified Batch will vary by task type. For more information, see the Specifying Task Parameters section in the OFS Analytical Applications Infrastructure User Guide.
- **Refresh**: Click **Refresh** to refresh the Simplified Batch summary page, including updates to the Last Execution Date and Status.

3. To select a Simplified Batch Rule, select a check box in the first column of the table. More than one Simplified Batch Rules can be selected at a time, but this disables some of the action icons.
   You can select or deselect all the Simplified Batch Rules in the summary table by selecting the check box in the upper left-hand corner of the summary table directly to the left of the Name column header.

   The following list provides the Simplified Batch Rule details based on the search criteria.

   - **Name**: Displays the Simplified Batch rule's short name. Hover on a batch name displays the rule's system identifier, long name, and description.

     > **Note**
     >
     > The rule's long name consists of a concatenation of the Information Domain (Infodom) and the rule's short name. This long name is the name under which the batch is submitted internally through the Oracle Financial Services Analytical Applications (OFSAA) Infrastructure.

   - **Creation Date**: Displays the Date and Time at which a Simplified Batch rule was created.
   - **Created By**: Displays the name of the user who created a Simplified Batch rule.
   - **Last Modification Date**: Displays the Date and Time at which a Simplified Batch rule was last modified.
   - **Last Modified By**: Displays the name of the user who last modified a Simplified Batch rule.
   - **Folder**: Displays the name of the Folder in which a Simplified Batch rule is stored.
   - **Last Execution Date**: Displays the Date and Time at which a Simplified Batch was last executed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Status**: Displays the latest status for a batch rule and includes the following possible values: Complete, Failed, New, Ongoing, Cancelled

8.2 Simplified Batch Details

When you Add, Edit, or View a batch, the Simplified Batch details window is displayed. The Simplified Batch details window presents the following panes for defining and maintaining your batch:

- Simplified Batch - header level information, such as Rule Name, and so on
- Batch Execution Type
- Search Task Details
- Task Details
- Audit Trail

When you **Add**, **Edit**, or **View** a Simplified Batch Rule, the **Simplified Batch Rule** details window is displayed. The Simplified Batch Rule details window includes a Simplified Batch Rule details pane, three Simplified Batch Rule panes, and an **Audit Trail** pane.

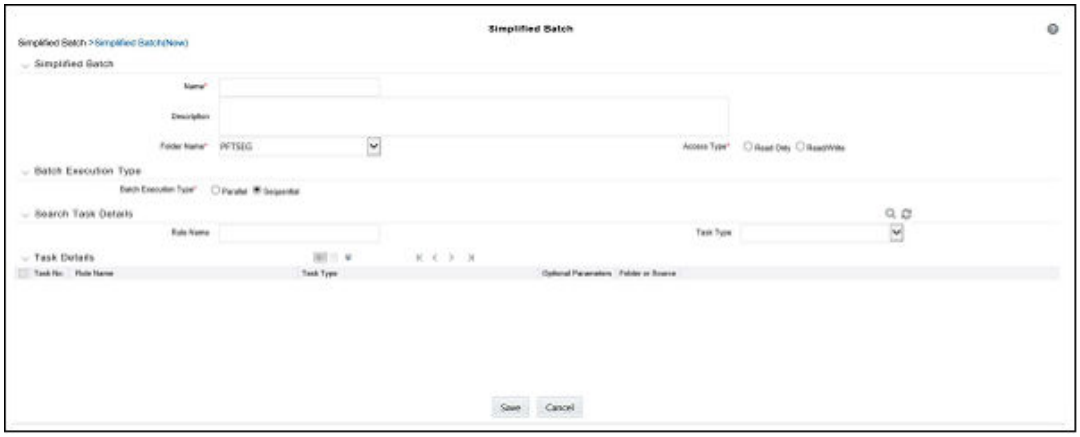
The Audit Trail pane is a standard footer pane for every OFSAA rule type. It displays Created By, Creation Date, Last Modified By, and Modification Date. The **User Comments** tab can be used to add comments to any rule, subject to a maximum of 4000 characters.

8.3 Creating a Simplified Batch

To create a Simplified Batch Rule, follow these steps:

1. Click Add from the Simplified Batch Rule summary page.

Figure 8-3 Simplified Batch details window to create a new Simplified Batch Rule



2. Enter the following information in the Simplified Batch Rule details window:
   - **Name * and Description**: Simplified Batch Name and Description.
     You cannot edit the Name after you have created the rule.

     The maximum length for the Name is based on (name and Infodom), subject to the total length not exceeding 42 characters. Underscore (_) is the only special character allowed for a Simplified Batch name.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The maximum length for the Description is 300 characters. Allowed special characters for a Simplified Batch description are underscore (-), hyphen (-), Colon (:), Dot (.), and Space.

- **Folder Name***: Select a folder in which the batch is to be stored.
- **Access Type***: Specify whether you want the batch rule to be Read, Write, or Read-Only.
  *** indicates Mandatory UI element.**

3. Click **Save**. To build out your Simplified Batch Rule, enter the data within the Simplified Batch Rule details page. For more information, see the following sections:
   - Batch Execution Type
   - Searching Task Name
   - Task Details
   - Task Selection

- Batch Execution Type
  Select the tasks in the batch to run in Parallel or Sequentially. To run some tasks sequentially and some in parallel, utilize the OFSAA Infrastructure Operations batching framework to construct the batch. The Simplified Batching Framework provides a simpler and faster pathway to specify many of the batches that require, but the OFSAA Infrastructure Operations framework offers more flexibility.
- Searching Task Name
  You can construct batches containing dozens or even hundreds of tasks. In such cases, it can become difficult to isolate a specific task.
- Task Details
  For a new batch rule, the Task Details pane will initially be empty. Use the Task Selector to populate the Task Details pane with the processes to run in the batch. For a Sequential batch, the Task Details pane presents the tasks in the order in which they will be run. For a Parallel batch, all tasks are initiated in parallel.
- Task Selection
  Click **Task Selector** from the **Task Details** pane to open the **Task Selector** window to support the maintenance of tasks in a batch.

8.3.1 Batch Execution Type

Select the tasks in the batch to run in Parallel or Sequentially. To run some tasks sequentially and some in parallel, utilize the OFSAA Infrastructure Operations batching framework to construct the batch. The Simplified Batching Framework provides a simpler and faster pathway to specify many of the batches that require, but the OFSAA Infrastructure Operations framework offers more flexibility.

Figure 8-4 Batch Execution Type on Simplified Batch details Window



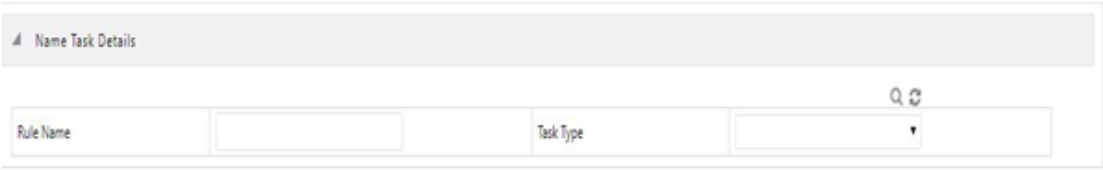
Copyright © 2011, 2026, Oracle and/or its affiliates.

8.3.2 Searching Task Name

You can construct batches containing dozens or even hundreds of tasks. In such cases, it can become difficult to isolate a specific task.

The Search Task details pane allows you to focus on the Task Details and display a specific task name or task type. The Rule Name search is a wild card character search. Enter **Task Type**, **Folder**, or **Task Name**, and then click **Search**. Click **Refresh** to remove the search criteria and refresh the Task Details pane to display all tasks within the batch.

Figure 8-5 Searching for Task Details



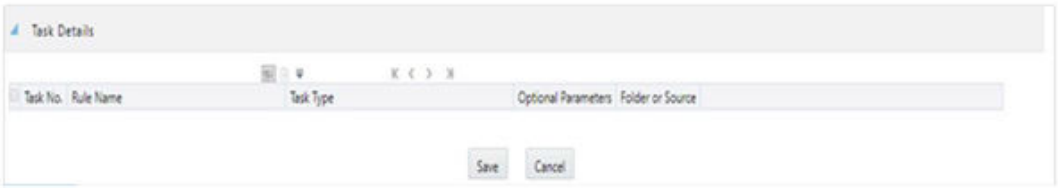
8.3.3 Task Details

For a new batch rule, the Task Details pane will initially be empty. Use the Task Selector to populate the Task Details pane with the processes to run in the batch. For a Sequential batch, the Task Details pane presents the tasks in the order in which they will be run. For a Parallel batch, all tasks are initiated in parallel.

The Task Details pane also provides a **Delete** icon for removing tasks from the batch.

- **Delete**: Selecting one or more rows in the Task Details page enables the Delete icon. Click Delete to remove the task or tasks that you have selected. You can also remove tasks through the Task Selection window.
- **Task Selector**: The Task Selector icon is always enabled. Click the Task Selector to open the Task Selection window.

Figure 8-6 Task Details pane of Simplified Batch details Window



This screen displays the following fields:

- **Task Number**: Each task is arranged sequentially by the task number.
- **Rule Name**: Displays the task's short name, such as an Allocation rule name. The complete or long name under which the task is submitted for execution (becoming the task's Batch Run ID) is a concatenation of several identifiers, as noted in the View Log. For more information, see the Viewing the Task Logs section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

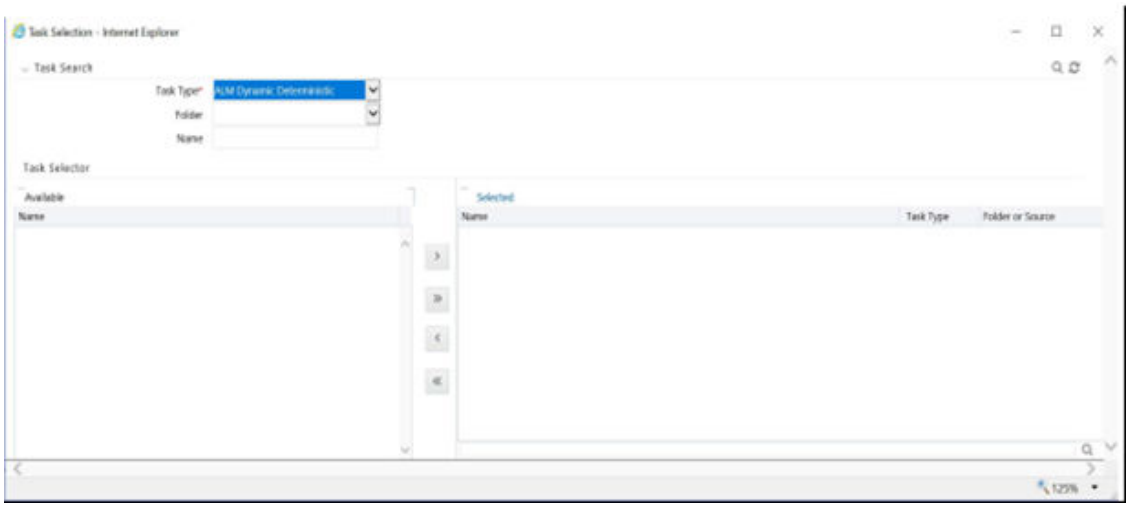
- **Task Type**: Use Simplified Batches to submit the different tasks for each of the application engines and various other components. For ALM, the following task types can be submitted within a Simplified Batch:
  - ALM Dynamic Deterministic
  - ALM Dynamic Stochastic/Historical Simulation
  - ALM Static Deterministic
  - ALM Static Stochastic/Historical Simulation
  - Cash Flow Edits
  - Extract Data
  - RUN DQ RULE
  - SQL Rule
  - Transform Data

  This list will vary based on the installed Application pack (or packs).
  - **As of Dates and Additional Parameters**: The As of Date applicable to each task within a Simplified Batch will vary by task type. Some task types also accept runtime parameters. For more information about the task parameters, see the Specifying Task Parameters section in the OFS Analytical Applications Infrastructure User Guide.
  - **Folder or Source**: For Extract Data tasks, values in this column represent Sources. For Transform Data tasks, this column is not applicable and is left blank. For all other task types, values in this column represent the Folder in which a rule is stored.

8.3.4 Task Selection

Click **Task Selector** from the **Task Details** pane to open the **Task Selector** window to support the maintenance of tasks in a batch.

Figure 8-7 Task Selection window



Use the **Task Selection** window to perform the following:

- Choose tasks to include in your batch

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Remove tasks from your batch
- Reorder tasks within your batch

To select the task from the **Task Selector** window, follow these steps:

1. After opening the **Task Selection** window, the list of **Available Tasks** on the left-hand side of the **Task Selector** window is blank.
2. To populate the list of available tasks, search the task in the **Task Search** pane. You must select a specific type of task from the Task Type drop-down list, and optionally specify search criteria for Folder (or Source) and Name, where applicable to the task type, and click Search.
3. Select the Task Type.
   For ALM, the following are the available task types:
   - ALM Dynamic Deterministic
   - ALM Dynamic Stochastic/Historical Simulation
   - ALM Static Deterministic
   - ALM Static Stochastic/Historical Simulation
   - Cash Flow Edits
   - Extract Data
   - RUN DQ RULE
   - SQL Rule
   - Transform Data

   This list will vary based on the installed Application pack (or packs).
4. Select the **Source** or **Folder**.
   - **Extract Data Task**: Select the Extract Data task to select a data source before executing your search.
   - **Transform Data Task**: After selecting the Transform Data task, do not select a Folder before executing the search.
   - **All Other Task Types**: Select any other Task Type to constrain the search to a particular folder. Alternatively, search for the tasks in all Folders by selecting blank (all folders).

   Executing a search for tasks always refreshes the **Available** list of tasks, but never has any impact on tasks that you have already **Selected** (tasks on the right-hand side of the window).

   You can further refine the search by entering a **Task Name**. Search using **Name** is wild card character searches, that is, searching for Name-like rate will find all tasks that include rate within their Name.
5. For Task Types with potentially large volumes of tasks, the **Available** list is divided into pages. Use the vertical scrollbar to navigate to the bottom of each page and click more to display the names on the next page. You can control the number of tasks listed per page by changing the setting in the `SIMPLE_BATCH_PAGE_SIZE` parameter in the `FusionApps.properties` file in the `WEB-INF/props` directory under context root. The recommended default value is 50 records per page. As you increase the value over a certain amount, you will start to see a slower response in the moving tasks between the Available and Selected lists.
6. Move the tasks between the Available and Selected lists.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Use the **Select** or **Select All** buttons to move rules from the displayed page of the Available list to the Selected list or vice versa. The Select ( ) moves rules from the left to the right, adding rules from the Available list to the bottom of the Selected list. Moving rules from the right to the left remove rules from the Selected list. You can also use the drag-and-drop feature to place the rules in a specific position in the Selected list to specify the execution order for a Sequential Type batch.

Tasks within the **Available** list of tasks are disabled (greyed-out) after they are selected (that is, included in the **Selected** list of tasks).

The **Task Selector** also supports drag-and-drop operations for choosing Available tasks, removing Selected tasks, or rearranging the sequence of the Selected tasks.

- Select an **Available** task and then drag-and-drop it anywhere within the Selected list, where the targeted placement is displayed in the red line.
- Dragging a task from the **Selected** list and dropping it into the **Available** list removes that task from the **Selected** list.
- To select or deselect a range of tasks, click the first task in the range to be selected, press the Shift key, click the last task in the range to be selected, and then drag-and-drop the set of tasks.
- To select or deselect a set of tasks that are not in sequence, click each task individually while pressing the Control key, after clicking on the last task to drag-and-drop the set of tasks.

7. Find a Selected task.
   To search for any of the selected tasks, enter the Task Name or partial name in the field under the Selected list, and then click Search.

8.4 Running a Simplified Batch

To run a Simplified Batch from the Simplified Batch summary page, follow these steps:

1. Select a Simplified Batch from the **Simplified Batch** summary page.
2. Click **Run** to execute the batch.
3. A confirmation message is displayed. Click **Yes** to proceed, or **No** to return to the Simplified Batch summary page.
4. After clicking **Yes**, a **Date Selection** window is displayed. Select the appropriate date for use in processing. The Date defaults to the System Date. To change the Date, either directly enter a date in the format displayed, or use the **Calendar** tool. You can ignore the message *Please select a date if you have included Extract Data or Transform Data tasks in this batch*.
5. Click **OK**.
6. A confirmation message is displayed. Click **OK**.
7. You can monitor the status from the Simplified Batch summary page or the Simplified Batch details page.
   a. To monitor the status from the Simplified Batch summary page, click **Refresh**. The summary page will update the Last Execution Date with the current date and the Status at completion (for example, Complete, or Failed if there was a critical problem).
   b. To monitor the status from the Simplified Batch details page, see the Viewing the Task Logs section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> This currently applies to all task types except Extract Data and Transform Data.

8.5 Viewing the Task Logs

To view the processing log for each task in the batch, navigate to the log details from the Simplified Batch details page:

1. Select the batch from the **Simplified Batch** summary page.
2. Click **View** to navigate to the Simplified Batch details page.
3. For a specific task in the **Task Details** pane, navigate to the processing log through View Log.
4. In the **View Log** window, view the log details through the **Task ID** link.
   The Task ID represents the System ID of the task (for example, in Profitability Management, an Allocation Rule's System ID).

   The task's Batch Run ID is a concatenation of several identifiers. For example, Batch Run ID **OELPMINFO_Task1_smb alloc2_20110805_2** represents the following:

   - The information Domain name (for example, OELPMINFO)
   - Task sequence within the Simplified Batch (for example, Task1)
   - Simplified Batch rule name (for example, smb alloc2)
   - The run date in YYYYMMDD format and the sequential run number for that date (for example, 20110805_2, for the second execution on August 5, 2011).

8.6 Additional Batch Options

You can also view, execute, schedule, monitor, cancel, and view task logs for Simplified Batches within the OFSAA Infrastructure Operations batching framework.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9

OFSAA Rate Management

OFSAA Rate Management is a comprehensive utility enabling you to manage currencies, yield curves, interest rates, and currency exchange rate data with a high degree of security and. OFSAA Rate Management also allows you to maintain economic forecasts such as GDP growth, inflation rates, or unemployment projections that can be linked to your models for interest rates, exchange rates, or new business growth.

Historical rate data obtained from OFSAA Rate Management is utilized by the Enterprise Performance Management (EPM) applications (OFS Funds Transfer Pricing, OFS Profitability Management, OFS Asset Liability Management, OFS Balance Sheet Planning, and OFS Cash Flow Engine).

- Interest Rates
- Currency
- Currency Rates
- Economic Indicators
- Interest Rates
  The quality and availability of interest rate information vary 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. The Interest Rates module in OFSAA Rate Management allows you to define and manage complex yield curve definitions using multiple rate formats and other rate attributes to give you data storage capabilities appropriate to your market. The Interest Rates module also supports the creation and maintenance of historical rate data for each yield curve you define.
- Interest Rates Details
  When you **Add**, **Edit**, or **View** an interest rate curve, the **Interest Rate Code Details** window is displayed. The **Interest Rate Code Details** window includes an **Interest Rate Code Details** pane, five **Interest Rate Code** tabs, and an **Audit Trail** pane.
- Currency
  Financial institutions transact business in more than one currency. Transacting business in multiple currencies demands functional capabilities for multi-currency accounting and currency rate management.
- Currency Rates
  OFSAA Rate Management's Currency Rates module uses the currencies defined and activated in the Currency module to support the creation and maintenance of historical exchange rates. Currency exchange rates are utilized in:
- Currency Exchange Rate Validation
  Exchange Rate Validation has the following features:
- Economic Indicators Summary Page
  An economic indicator is any economic statistic such as the Consumer Price Index (CPI), growth rate of the Gross Domestic Product (GDP), unemployment rate, Purchasing Managers Index, indices of consumer confidence, and so on. Such macroeconomic statistics tell us how well the economy has behaved in the past. Some economic indicators are referred to as lagging indicators while others are classified as leading indicators. Leading indicators can provide insights into the future direction of the economy.

Copyright © 2011, 2026, Oracle and/or its affiliates.

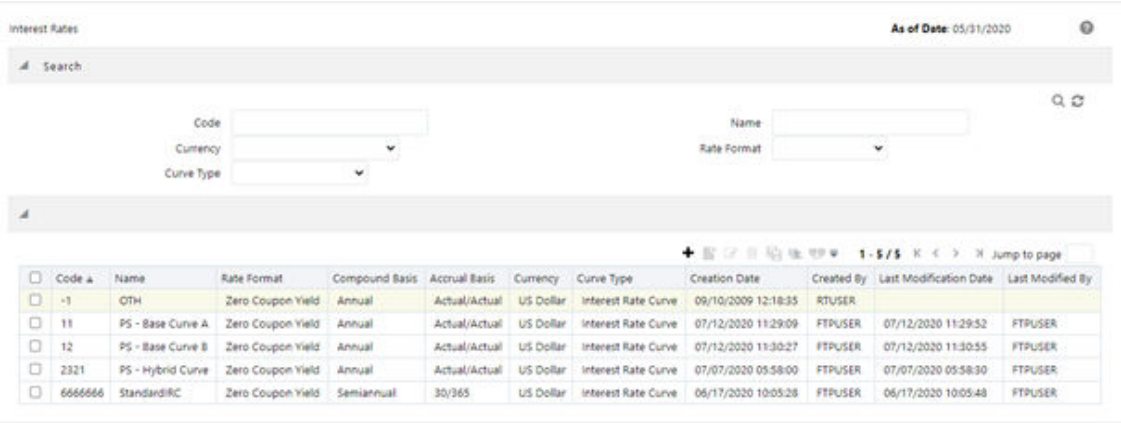
9.1 Interest Rates

The quality and availability of interest rate information vary 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. The Interest Rates module in OFSAA Rate Management allows you to define and manage complex yield curve definitions using multiple rate formats and other rate attributes to give you data storage capabilities appropriate to your market. The Interest Rates module also supports the creation and maintenance of historical rate data for each yield curve you define.

Historical interest rate data from OFSAA Rate Management is utilized in OFSAA Transfer Pricing to generate transfer rates and option costs. Historical interest rate data is also utilized in OFSAA Asset Liability Management, OFSAA Balance Sheet Planning, and OFSAA Cash Flow Engine to generate forecasted interest rate scenarios.

To view the Interest Rate Code Summary page, navigate to Common Object Maintenance and select Rate Management, and then select Interest Rates.

Figure 9-1 Interest Rate Summary Page



- Searching for Interest Rate Curve
  A Search pane is provided to search for interest rate curves using Code, Name, Currency, Rate Format, or Curve Type parameters.

9.1.1 Searching for Interest Rate Curve

A Search pane is provided to search for interest rate curves using Code, Name, Currency, Rate Format, or Curve Type parameters.

To search the Interest Rate, follow these steps:

1. Enter the search criteria and click **Search**.
2. Click Reset to remove any **Code**, **Name**, **Currency**, **Rate Format**, or **Curve Type** constraint specified and refresh the window.
   The search results are displayed in a table containing all of the interest rate curves that meet the search criteria. The **Interest Rates Summary** page offers several icons that allow you to perform different functions when an interest rate curve is selected.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Add**: Click Add to build a new interest rate curve. The Add icon is disabled if any row in the pane is selected.
- **View**: Select a single row in the table to enable the View icon. Click View to view the contents of an Interest rate curve in read-only format. The View icon is enabled only when a single Interest rate curve is selected in the table.
- **Edit**: Select a single row in the table to enable the Edit icon. Clicking the Edit icon allows you to modify a previously saved Interest rate curve. The Edit icon is only enabled when a single row is selected. You can control the number of rows to display on the window by selecting the Pagination Options icon from the action bar.
- **Delete**: Select one or more rows out of the table to enable the Delete icon. Clicking on the Delete deletes the Interest rate curve you have selected.
- **Copy**: Selecting a single row in the table enables the Copy icon. Click the Copy icon to create a copy of an existing Interest rate curve. The Copy icon is only enabled when a single Interest rate curve is selected.
- **Check Dependencies**: Select an interest rate curve and then click the Check Dependencies icon to generate a report on all rules that utilize your selected interest rate curve. The Check Dependencies icon is only enabled when a single interest rate curve is selected.
- **Execute the Historical Interest Rates Data Loader**: This icon executes a function to import historical rates and parameters for all defined Interest rate curves. For more information on setting up the automated process, see the OFS Data Model Utilities User Guide.
  `FusionApps.properties` file must be present under Web Server `<deployed path>/WEB-INF/props` and `$FIC_WEB_HOME/webroot/WEB-INF/props` directories.

  Update the entry details of the source name. It must be `infodom_SOURCE_NAME=Actual_Source_Name`.

  For example, if Infodom Name is `DEV6INFO` and the source Name is `DEVETLSRC`, then the entry will be `DEV6INFO_SOURCE_NAME=DEVETLSRC`.

  Launch the Historical Interest Rates Data Loader from the Interest Rates Summary page after making this change.

  For more information, see Doc ID 2233513.1.

  To execute a historical Interest Rate data load, execute the **Historical Interest Rates Data Loader**. A warning message will appear *Upload all available Interest Rates and Parameters?* Click **Ok** to load all historical rates and parameters.

  Click **Data Loader** to execute all the interest rates and parameters.

3. To select an Interest rate curve, select a check-box in the first column of the table. More than one Interest rate curve can be selected at a time, but this disables some of the icons. You can select or deselect all the Interest rate curves in the Summary page by selecting the check box in the upper left-hand corner of the Summary table directly to the left of the Code column header.

   The following list provides the Interest Rate details based on the search criteria.

   - **Code**: Displays the interest rate curve's code. The code is a unique number in the range of 1 to 9999999. Hover on a row in the pane to display the interest rate curve's detailed description.
   - **Name**: Displays the interest rate curve's short name.
   - **Rate Format**: Displays the Interest rate curve's rate format (zero-coupon or yield-to-maturity).

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Compound Basis**: Displays the Interest rate curve's compounding basis (Annual, Semiannual, Monthly, or Simple).
- **Accrual Basis**: Displays the Interest rate curve's Accrual Basis (that is, 30/360, Actual/Actual, and so on).
- **Currency**: Displays the Currency (Reference Currency) to which an Interest rate curve is applicable.
- **Curve Type**: Displays the curve type as an Interest rate curve or Volatility Curve.
- **Creation Date**: Displays the Date and Time at which an Interest rate curve was created.
- **Created By**: Displays the Name of the user who created the Interest rate curve.
- **Last Modification Date**: Displays the Date and Time at which an Interest rate curve was last modified.
- **Modified By**: Displays the Name of the user who last modified the Interest rate curve.

9.2 Interest Rates Details

When you **Add**, **Edit**, or **View** an interest rate curve, the **Interest Rate Code Details** window is displayed. The **Interest Rate Code Details** window includes an **Interest Rate Code Details** pane, five **Interest Rate Code** tabs, and an **Audit Trail** pane.

The **Audit Trail** pane is a standard footer pane for every OFSAA rule type. The **Audit Trail** pane displays **Created By**, **Creation Date**, **Last Modified By**, and **Modification Date**. The **User Comments** tab can be used to add comments to any rule, subject to a maximum of 4000 characters.

- Creating an Interest Rate Code
  To create an Interest Rate Code, follow these steps:
- Interest Rate Code Tabs
  Interest Rate Code tabs are used to define the yield curve and to add, edit, or delete historical interest rate data. The Interest Rate Code tabs are:
- Batch Creation
  There are two methods for creating and running the batch processes, ICC Batch and Simplified Batch. The following section describes how to set up and run the Hybrid Rate Generation using both approaches. For more information, see the OFS Analytical Applications Infrastructure User Guide.

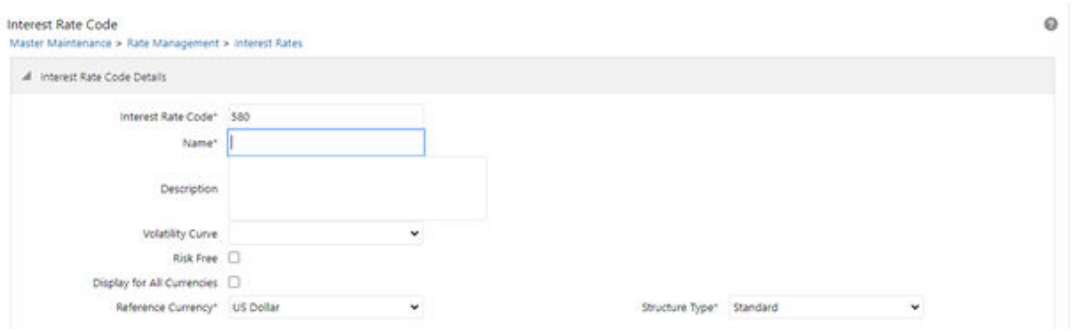
9.2.1 Creating an Interest Rate Code

To create an Interest Rate Code, follow these steps:

1. Click **Add** from the Interest Rate Code Summary page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 9-2 Interest Rate Code Details window



2. Enter the following information in the Interest Rate Code window.

   - **Interest Rate Code**: When constructing a new yield curve, you must specify an Interest Rate Code between 1 and 9999999. Interest Rate Codes are used internally to uniquely identify yield curves. When working with Rate Management or other OFS Analytical Applications, you reference yield curves by Name, not by Interest Rate Codes. Interest Rate Codes are embedded within your instrument data (for example, the INTEREST_RATE_CD and T_RATE_INT_RATE_CD columns within the instrument data are populated with Interest Rate Codes). After you have saved a yield curve, you cannot renumber its Interest Rate Code.
   - **Name and Reference Currency**: You must provide a Name and Reference Currency for your yield curve. Unlike Interest Rate Codes, you can rename or change the Reference Currency for previously saved yield curves. While you can choose to rename a yield curve, however, it is very unlikely that you will choose to modify a yield curve's Reference Currency. A yield curve's Reference Currency is the currency for which your market rates are valid. For example, the Reference Currency for a Prime Rate yield curve would be US Dollars. LIBOR or other internationally quoted rates are always quoted with respect to an underlying Reference Currency (for example, US Dollar LIBOR, Euro LIBOR, and so on). The Reference Currencies drop-down list displays only Active currencies. For more information on Active and Inactive currencies, see the Currency section.
   - **Description**: You can optionally describe or modify your yield curve's description at any time.
   - **Structure Type**: This attribute is required for each yield curve. Structure Type supports both Standard and Hybrid yield curve definitions. Hybrid yield curves are re-expressions of one or more pre-existing Standard yield curves. For more information, see Hybrid Term Structure Tab under Interest Rate Code Tabs. After you have saved the yield curve, you cannot change the selected Structure Type.
   - **Volatility Curve**: You can select the Volatility Curve option to indicate the curve selected will contain volatility rates. If you select this option, all other curve attributes become disabled and the curve is used exclusively for managing volatility details. For FTP Volatility, both the Terms tab and Historical Rates tab are available with this option. For ALM Volatility, only the Historical Rates tab is available. After you have saved the yield curve, you cannot change the selected Volatility Curve. Volatility curves are used in FTP to calculate the Rate Lock Option Costs. ALM Volatility is used to evaluate embedded options for the Black 76 market valuation.
   - **Risk-Free**: (Optional) This flag is for tagging IRCs as risk-free. That is editable in new and edit modes. It is available for non-hybrid curves and hybrid curves, and not available for Volatility Curves.
   - **Display for all currencies**: This option allows you to designate certain Interest rate curves to make them available for assumption mapping to any currency. Assumption

Copyright © 2011, 2026, Oracle and/or its affiliates.

rules filter the list of Interest Rate Codes based on the currency when defining assumptions for a specific product/currency combination. When this option is enabled, the Interest Rate Code appears in assumption rules for all currencies.

3. Click **Save**. To build out the Interest Rate Code, you must enter data within the Interest Rate Code tabs. For more information, see the Interest Rate Code Tabs section.

9.2.2 Interest Rate Code Tabs

Interest Rate Code tabs are used to define the yield curve and to add, edit, or delete historical interest rate data. The Interest Rate Code tabs are:

- Attributes
- Terms
- Historical Rates
- Parameters
- Hybrid Term Structure

- Navigating Between Interest Rate Code Tabs
  For new yield curves, you must begin with the Attributes tab. After you have selected attributes for a yield curve, you cannot edit them. After assigning the attributes, navigate to the Terms tab to define a term structure for your yield curve or volatility curve, that is, an overnight rate, a one-month rate, a three-month rate, and so on. To navigate to the Terms tab, either click Apply on the Attribute tab or click the Terms tab.
- Parameters Tab
  Fixed income instruments are used for forecasting and simulating the Cash Flows. The Cash Flow Engine needs interest rate models to simulate the evolution of interest rates. The Cash Flow Engine uses these models as part of the stochastic engine. You can enter the parameters for these models in the following ways:
- Hybrid Term Structure Tab
  Hybrid Term Structures allows you to specify the following three types of Hybrid yield curves:

9.2.2.1 Navigating Between Interest Rate Code Tabs

For new yield curves, you must begin with the Attributes tab. After you have selected attributes for a yield curve, you cannot edit them. After assigning the attributes, navigate to the Terms tab to define a term structure for your yield curve or volatility curve, that is, an overnight rate, a one-month rate, a three-month rate, and so on. To navigate to the Terms tab, either click Apply on the Attribute tab or click the Terms tab.

> **Note**
>
> You must specify an Interest Rate Code, Name, and Reference Currency in the Interest Rate Code Details window before navigate to the Terms tab.

The first time you navigate to the Terms tab, an initial 1-month term point is provided, but even if this is the only term point you want for the curve, you must click Apply to finish the term structure specification. In future revisions to the curve's definition, navigate directly to the Historical Rates tab, but if you modify the term structure, you must always click Apply on the Term tab before navigating to the Historical Rates tab. The Historical Rates tab is used to input interest rate or volatility data. This tab is used for maintaining the interest rates database. To

Copyright © 2011, 2026, Oracle and/or its affiliates.

navigate to the Historical Rates tab, either click Apply on the Terms tab or select the Historical Rates tab if you have already defined your term structure.

> **Note**
>
> You must specify the following before navigating to the Historical Rates tab:
>
> - An Interest Rate Code, Name, and Reference Currency in the Interest Rate Code Details window.
> - A term structure in the Terms tab.

- Attributes Tab
  Yield curve attributes include Rate Format, Compounding Basis, Accrual Basis, and Curve Identifier.
- Terms Tab
  Use the Terms tab to construct the yield curve's term structure. You can specify as many yield curve terms from the 1 day to 100 years range.
- Historical Rates Tab
  Use the Historical Rates tab to define, modify, or view interest rate data. Enter data in simple percentages (such as 5.125, 4.875, and so on). If you are entering historical rates for a volatility curve, then enter volatilities for the combination of Contract Term and Expiration Term. Select the desired Expiration Term from the drop-down list to complete the Volatility Matrix. Effective dates must be entered separately for each Expiration term in the list.
- Historical Rates Tab on Interest Rate Code window
  The Historical Rates Tab pane on the Interest Rate Code window allows you to see the historical rates available for the interest rate code that is being created.
- Add Historical Rates
  By default, the Historical Rates tab displays interest rate data for the past month (for example, for the 30 days leading up to the current date). Click the Effective Date Range drop-down list to expand your view to the last 3 months, 6 months, one year, 3 years, 6 years, or all rate data.
- Rate Lookup Behavior between Term Points
  The OFS Cash Flow Engine is common to OFS FTP, OFS PMTPC, OFS ALM, and OFS BSP applications. To lookup rates from Rate Management, the Cash Flow Engine performs an interpolation between yield curve Term Points. For example, in determining a Straight Term Transfer Rate (common for products such as time deposits), the Engine must determine a three-month rate from a yield curve that contains only a one-month rate and a six-month rate. In such a case, the Cash Flow Engine performs interpolation to determine the implied three-month rate. While each of the applications supports simple linear interpolation, OFS FTP and OFS ALM also support Cubic and Quartic Spline interpolation methods. These more advanced methods will be supported for all the OFS Analytical Applications in a future release.
- Rate Lookup Behavior Beyond Term Points
  If the Cash Flow Engine must determine a rate from a yield curve for a term point smaller than the first term point of the yield curve, then the Engine utilizes the first term point of the yield curve. For example, if the Engine must determine an overnight rate from a yield curve whose first term point is one month, the Engine utilizes the one-month rate.

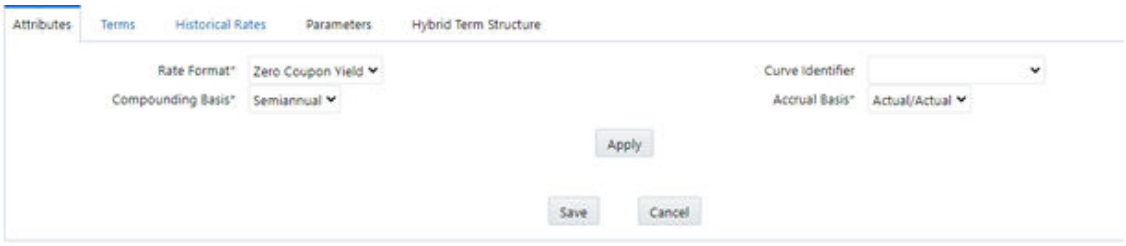
Copyright © 2011, 2026, Oracle and/or its affiliates.

- Rate Lookup Behavior between Effective Dates
  When you are looking up rates from Rate Management for a business date, the Cash Flow Engine helps to find if there is no rate data for that specific business date.
- Deletion of Historical Rates
  To delete historical rates entered, select one or more rows and then click **Delete**.
- Generate Graph
  The Generate Graph option allows you to generate the graph for selected Historical rates.
- Excel Import or Export
  To aid in data entry, use the Excel Import or Export functionality to add or edit rate data to historical rates. This is an optional step.

9.2.2.1.1 Attributes Tab

Yield curve attributes include Rate Format, Compounding Basis, Accrual Basis, and Curve Identifier.

Figure 9-3 Attribute Tab of Interest Rate Code screen



All attributes are disabled if the Volatility Curve is selected. The following list describes the fields in the Attribute Tab of Interest Rate Code screen.

- **Rate Format**: To define the yield curve, you must select either the Zero Coupon or Yield to Maturity Rate Format. Rates entered into Rate Management (in the Historical Rates tab) are always entered in the nominal form, such as 5.125% or 6.875%, not as discount factors. For more information on how the two rate formats affect internal cash flow engine calculations, see the OFS Cash Flow Engine Reference Guide.
- **Compounding Basis**: Select a Compounding Basis for the yield curve: Annual, Semiannual, Monthly, or Simple. Annual is the most common method. The Monthly Option is enabled based on Rate Format selection (if Rate Format is selected as Yield to Maturity). For more information on Compounding Basis and how different compounding bases affect cash flow calculations in OFSAA, see the OFS Cash Flow Engine Reference Guide.
- **Accrual Basis**: Select an Accrual Basis for the yield curve. The Accrual Basis list depends on the Compounding Basis selection. If the Compounding Basis is selected as Annual, Semiannual, or Monthly, then the following Accrual Basis types are available:
  - Actual/Actual
  - 30/365
  - Actual/365

  If the Compounding Basis is selected as Simple, then the following Accrual Basis types are available:

Copyright © 2011, 2026, Oracle and/or its affiliates.

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

For more information on Accrual Basis and how different accrual bases affect cash flow calculations in OFSAA, see the OFS Cash Flow Engine Reference Guide.

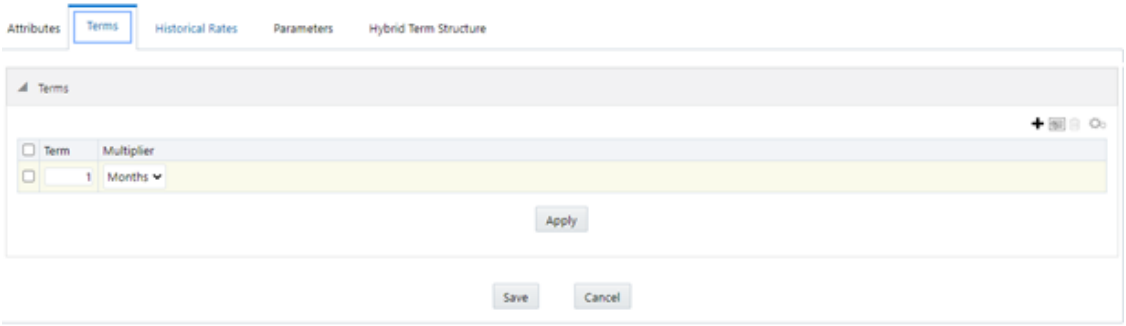
- **Curve Identifier**: The curve identifiers can be mapped to the Adjustable Rate Mortgage (ARM) indices used for ADCO prepayment model processing. This is a drop-down list that contains the six Curve Identifier types, that is, Par Treasury, Zero-coupon Treasury, Par LIBOR/Swap, Zero-coupon LIBOR/Swap, Prime Lending Rate, and Cost of Funds Index. The Reference Currency for these indices is always US Dollar.
  The curve identifier is optional when the IRC is not used for ADCO prepayment model processing. This is applicable only in ALM and HM applications.

9.2.2.1.2 Terms Tab

Use the Terms tab to construct the yield curve's term structure. You can specify as many yield curve terms from the 1 day to 100 years range.

> **Note**
>
> You must supply input data for all term points of an IRC. This applies to historical rate and direct input forecast method. If data of one or more terms points is not given then it can lead to errors during calculation.

Figure 9-4 Terms Tab on Interest Rate Code window



> **Note**
>
> When constructing a volatility curve, the term types available are Contract Term and Expiration Term. You must add the relevant terms for each of these dimensions. Click **Apply** after defining terms for each dimension to save the assumptions before proceeding.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Adding New Term Points**: Click **Add** to add new term points by selecting a Term value and a Multiplier (such as 7 days, 2 months, 5 years, and so on). You can also add term points using the Data Input Helper option. For more information, see the **Data Input Helper** section.
- **Deleting Existing Term Points**: To delete an existing term, select the term point (or terms), and click **Delete**. A confirmation message is displayed. Click **Ok**.

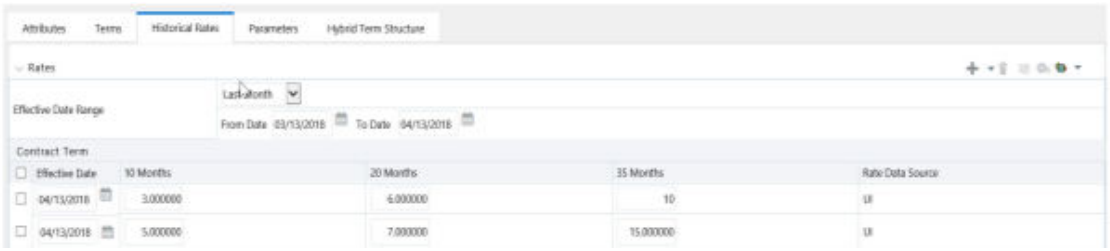
9.2.2.1.3 Historical Rates Tab

Use the Historical Rates tab to define, modify, or view interest rate data. Enter data in simple percentages (such as 5.125, 4.875, and so on). If you are entering historical rates for a volatility curve, then enter volatilities for the combination of Contract Term and Expiration Term. Select the desired Expiration Term from the drop-down list to complete the Volatility Matrix. Effective dates must be entered separately for each Expiration term in the list.

9.2.2.1.4 Historical Rates Tab on Interest Rate Code window

The Historical Rates Tab pane on the Interest Rate Code window allows you to see the historical rates available for the interest rate code that is being created.

Figure 9-5 Interest Rate Code window - Historical Rates Tab



> **Note**
>
> FTP Volatility curves are only applicable to FTP Rate lock option cost calculations and ALM/CFE Volatility only applies to embedded option market valuation.

To enter historical rates for an FTP volatility curve, enter volatilities for the Contract Term. Select the desired Expiration Term from the drop-down list to complete the Volatility Matrix. Effective Dates must be entered for each Expiration Term.

To enter historical rates for an ALM Volatility Surface, enter volatilities based on the following two dimensions:

- Strike Rate (Vertical Axis)
- Expiration Date (Horizontal Axis)

For a new ALM Volatility Surface, enter an As of Date and the number of breakpoints for Strike Rate and Expiration Date, then click Generate. The size of the matrix cannot exceed 20 x 20. For Strike Rate, enter values in ascending order. For Expiration Date, enter values in ascending order with values greater than the currently specified As of Date. At every intersection of Strike Rate and Expiration Date, enter a volatility amount in percent (that is, 25 = 25%).

Copyright © 2011, 2026, Oracle and/or its affiliates.

The Rate Data Source parameter shows from where the rates are taken from, either they are entered through the User Interface, loaded through the Data Loader, or generated using the Generate Rates of Hybrid IRC.

You can perform the following tasks:

- Add Historical Rates
- Rate Lookup Behavior Between Term Points
- Rate Lookup Behavior Beyond Term Points
- Rate Lookup Behavior Between Effective Dates
- Generate Graph
- Excel Import or Export
- Deletion of Historical Rates
- Data Input Helper

9.2.2.1.5 Add Historical Rates

By default, the Historical Rates tab displays interest rate data for the past month (for example, for the 30 days leading up to the current date). Click the Effective Date Range drop-down list to expand your view to the last 3 months, 6 months, one year, 3 years, 6 years, or all rate data.

9.2.2.1.6 Rate Lookup Behavior between Term Points

The OFS Cash Flow Engine is common to OFS FTP, OFS PMTPC, OFS ALM, and OFS BSP applications. To lookup rates from Rate Management, the Cash Flow Engine performs an interpolation between yield curve Term Points. For example, in determining a Straight Term Transfer Rate (common for products such as time deposits), the Engine must determine a three-month rate from a yield curve that contains only a one-month rate and a six-month rate. In such a case, the Cash Flow Engine performs interpolation to determine the implied three-month rate. While each of the applications supports simple linear interpolation, OFS FTP and OFS ALM also support Cubic and Quartic Spline interpolation methods. These more advanced methods will be supported for all the OFS Analytical Applications in a future release.

9.2.2.1.7 Rate Lookup Behavior Beyond Term Points

If the Cash Flow Engine must determine a rate from a yield curve for a term point smaller than the first term point of the yield curve, then the Engine utilizes the first term point of the yield curve. For example, if the Engine must determine an overnight rate from a yield curve whose first term point is one month, the Engine utilizes the one-month rate.

If the Cash Flow Engine must determine a rate from a yield curve for a term point greater than the longest term point on the yield curve, the Engine utilizes the last term point of the yield curve. For example, if the Engine must determine a 30-year rate from a yield curve whose last term point is 10 years, the Engine utilizes the 10-year rate.

9.2.2.1.8 Rate Lookup Behavior between Effective Dates

When you are looking up rates from Rate Management for a business date, the Cash Flow Engine helps to find if there is no rate data for that specific business date.

For example, in generating an original term transfer rate for an instrument with an origination date of June 14, 2010, the Cash Flow Engine may find rate data for May 31, 2010, and for June 30, 2010, but no rate data for any dates between May 31, 2010, and for June 30, 2010. In

Copyright © 2011, 2026, Oracle and/or its affiliates.

such cases, the Cash Flow Engine always falls back to the latest available rate data before the business date of interest (May 31, 2010, in this case).

9.2.2.1.9 Deletion of Historical Rates

To delete historical rates entered, select one or more rows and then click **Delete**.

9.2.2.1.10 Generate Graph

The Generate Graph option allows you to generate the graph for selected Historical rates.

To generate a graph, follow these steps:

1. Select the Effective Date Range on the Historical Rates tab.
   The From Date and To Date fields will be automatically updated after selecting the Effective Date Range.
2. Select the Effective Date using the corresponding **Calendar** icon.
3. Enter the term points in respective fields and click **Apply**.
4. Select the **Terms** using the corresponding check boxes.
5. Click **Generate Graph**. The graph is generated. Here, you can view the graph for Interest Rate vs Effective Date or Interest Rate vs Term Point.

9.2.2.1.11 Excel Import or Export

To aid in data entry, use the Excel Import or Export functionality to add or edit rate data to historical rates. This is an optional step.

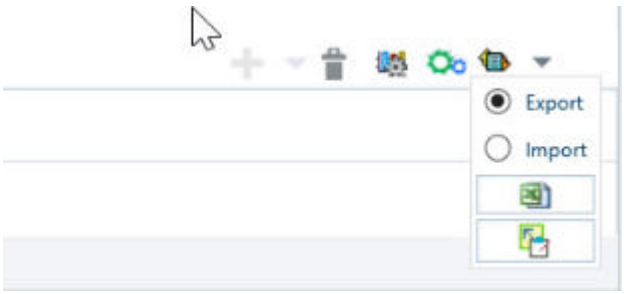
> **Note**
>
> Starting from the 8.0.4 release, the Excel Import feature will work only on that Internet Explorer browser. Excel Export feature is supported by all browsers such as Chrome or Firefox.

To export the data, follow these steps:

Excel Export:

1. On the Interest Rates toolbar, click the **Excel** icon.
2. Click **Export** to export data for the chosen selected effective date range. Within the same block, select **Export to Excel**, which launches the Excel application and displays the data window including headers.

Figure 9-6 Export and Import options for Historical Rates data



Copyright © 2011, 2026, Oracle and/or its affiliates.

Excel Import:

You can add rows to the pane in the same format. After complete data is entered (or existing data is edited), copy the range of the pane you want to append and go back to the **Historical Rates** pane. In the same block, click the **Import**. The data copied from excel will appear in the **Historical Rates** pane.

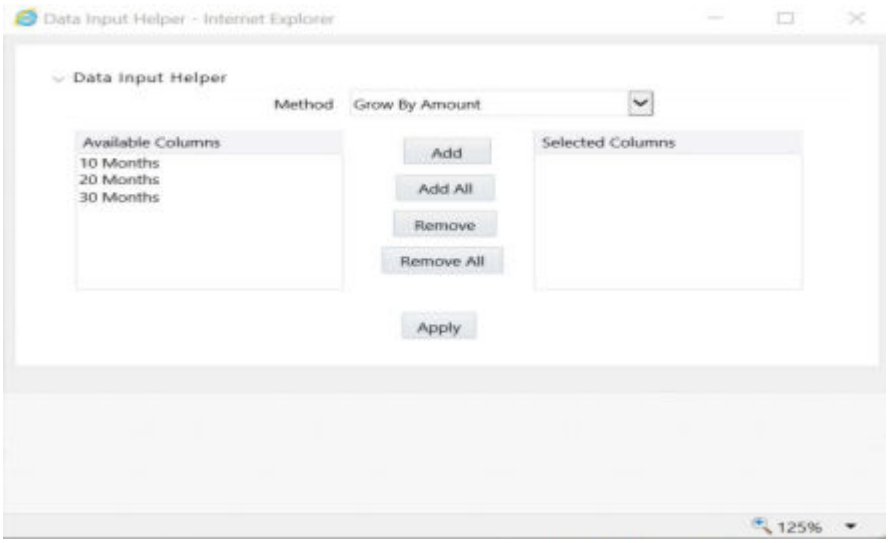
1. If appending data that pre-existed for the same effective date, the import will overwrite existing data.
2. In some cases, some fields will be output to the pane that is not editable, such as bucket start and end dates (when defining forecast assumptions). Add or edit data in the columns that would be editable only in the UI itself.

Data Input Helper:

Use the Data Input Helper feature to copy from a row where you have already defined the interest rate definition or apply a fixed value down the page. This is an optional step. To use data input helper, follow these steps:

1. Select the check box next to the rows that you want to work with or use the Select All option by selecting the check box on the header row.
2. Select **Data Input Helper**.
3. From the **Data Input Helper** window, select method - **Grow by Amount** or some other appropriate method.
4. Select the term points from the **Available Columns** list and click **Add** or **Add All** to move the data to the **Selected Column**s list.

Figure 9-7 Data Input Helper window



Based on the selected method, the list of Available Columns will vary. If Method is selected as Increment By Months, Increment By Days, or Increment By Years, then only the Effective Date option is displayed in the Available Columns list.

5. Click **Apply** to modify the Start date and enter the Incremental Value.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.2.2.2 Parameters Tab

Fixed income instruments are used for forecasting and simulating the Cash Flows. The Cash Flow Engine needs interest rate models to simulate the evolution of interest rates. The Cash Flow Engine uses these models as part of the stochastic engine. You can enter the parameters for these models in the following ways:

- System-generated calculations through Parameter Estimation
- Direct input into the UI
- Excel Import
- UI entry through Data Input Helper methods
- Data Loader

The following interest rate models are available:

- Extended Vasicek
- Ho and Lee
- Merton
- Vasicek
- Parameter Estimation
  This section explains the procedure to calculate the estimated parameters.

9.2.2.2.1 Parameter Estimation

This section explains the procedure to calculate the estimated parameters.

Prerequisites

Installation of R and Oracle R Enterprise (ORE) is required to use the Term Structure Parameter Estimation functionality under Rate Management - Interest Rates, for computing term structure parameters.

For more information, see the OFS Advanced Analytical Applications Infrastructure Installation and Configuration Guide.

Conditions for Parameter Estimation: If the following conditions are met, you can calculate parameters for any Term Structure Model for a given Effective Date, based on your relative look back term and a sufficient number of observations (available historical rates) for the IRC. If you rerun with a different look-back term, it will overwrite the existing parameters for the selected Term Structure Model on that IRC's Effective Date.

- **Term point**: Underlying historical rates must be available for a 30-Day or 1-Month term point.
- **Minimum Number of historical rates**: A total of at least 10 historical rates (observations) are required, on appropriate look-back dates.
- **Lookback Dates**: Historical rates must be available on dates looking back from the Parameter's Effective Date (the End Date), in 30-day intervals moving backward from End Date to Start Date, for a minimum of 10 intervals.
  For example: If the first rate's Effective Date is 1 Jan. 2013, then the second rate's date must be 2 Dec. 2012 (1 Jan. 2013, 30 days = 2 Dec. 2012), and so on. If a rate is not found for the required date, the engine looks for a rate within the neighborhood of 5 days up or down (therefore a total range of 10 days), searching iteratively starting with Date -1,

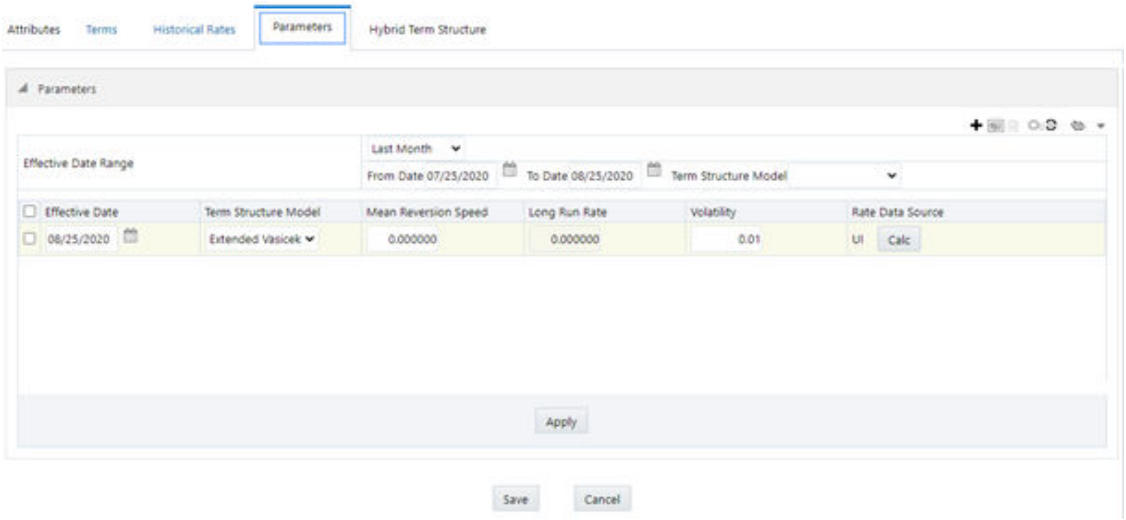
Copyright © 2011, 2026, Oracle and/or its affiliates.

then Date +1, through Date +5, then Date -5. The next rate lookup would be 60 days before the End Date, and so on.

The minimum relative term for all lookbacks must be at least 300 Days (that is, to accommodate a minimum of ten 30-day intervals). Using the above logic, if a rate is not found for the lookup date (or date within the neighboring range), an error will be logged in FSI_PROCESS_ERRORS with ID_Number to identify the Interest Rate Code, and the parameter estimation Engine will exit.

To define the Parameter Estimation, follow these steps:

Figure 9-8 Parameters Tab on Interest Rate Code window



To edit or recalculate the existing Parameters, enter the Effective Date Range filter.

1. The following icons are available here:
   - Add
   - Delete
   - Data Input Helper
   - Refresh
   - Excel Export or Import
2. Default parameters for the **Extended Vasicek Model** are displayed for one Effective Date (the System Date on which the Interest Rate Code was created). You can edit these parameters or add new parameters using **Add**. Use the **Refresh** to return to existing Parameters.

> **Note**
>
> Steps 3 and 4 are applicable only if you are Adding or Editing Parameters.

To add or edit the parameters, use the **Data Input Helper** or **Excel Import** or **Export**. For more information, see the Excel Import or Export and Data Loader sections.

Copyright © 2011, 2026, Oracle and/or its affiliates.

3. Enter the **Effective Date**. Note that the **Effective Date** cannot be greater than the **Current System Date**.
4. Select the Model from the Term Structure Model drop-down list. Effective Date and Term Structure Model combination must be unique within this IRC.
5. The following term structure models are utilized in stochastic modeling of interest rates in OFS FTP and OFS ALM:
   - Extended Vasicek
   - Ho and Lee
   - Merton
   - Vasicek
6. The following parameters needed by the models:

Table 9-1 List of supported parameters for Models Term structure models in Interest Rate

Model	Parameter 1	Parameter 2	Parameter 3
Extended Vasicek	Volatility	Mean Reversion Speed	
Ho and Lee	Volatility		
Merton	Volatility		
Vasicek	Volatility	Mean Reversion Speed	Long Run Rate

7. Values for Long Run Rate and Volatility are in percentages.
For example, a Long Run Rate of 5% is displayed as 5.000. To maintain the integrity of data, Rate Management restricts the accepted input values. The valid range and the default setting for each parameter.

Table 9-2 Valid Range and Default Values of Interest Rate Parameters

Parameter	Valid Range	Default Value
Volatility	0% to 500%	0.01
Mean reversion speed	0.00 to 500	0.0
Long run rate	0.00% to 500%	0.0

8. The **Rate Data Source** indicates if you have directly entered Parameters through the UI. You can calculate the Parameters to display system-generated values.
9. Select Calculate to view the Term Structures Parameter Estimation window.
10. If you are directly entering the Parameters, then enter values appropriate to the Model, then click **Apply** else click **Calculate**.
11. The End Date is auto-populated with the Effective Date. Enter the Relative Term with Multiplier to define the rate look-back period.
12. The Term or Multiplier value must be at least 300 Days.
13. The Start Date is automatically updated after entering a Relative Term.
14. Click **Calculate Number of Observations** to confirm the number of rates found for appropriate dates within the Relative Term. If there are at least 10 observations, then the **Estimate** option will become active.

Copyright © 2011, 2026, Oracle and/or its affiliates.

15. Click **Estimate** to calculate the parameters and store them in the Historical Parameters table (FSI_IRC_TS_PARAM_HIST). A confirmation message is displayed. Click **Ok**. The calculation will complete and you will be directed back to the Parameters tab.
16. Similar to the validation used for direct input from the UI, if any of the calculated parameters are outside of the valid range, the Engine displays an error message.

Delete parameters by selecting one or more rows and then clicking Delete. For more information on term structure models and stochastic processing, see the OFS Funds Transfer Pricing User Guide, OFS Asset Liability Management User Guide, and OFS Cash Flow Engine Reference Guide.

9.2.2.3 Hybrid Term Structure Tab

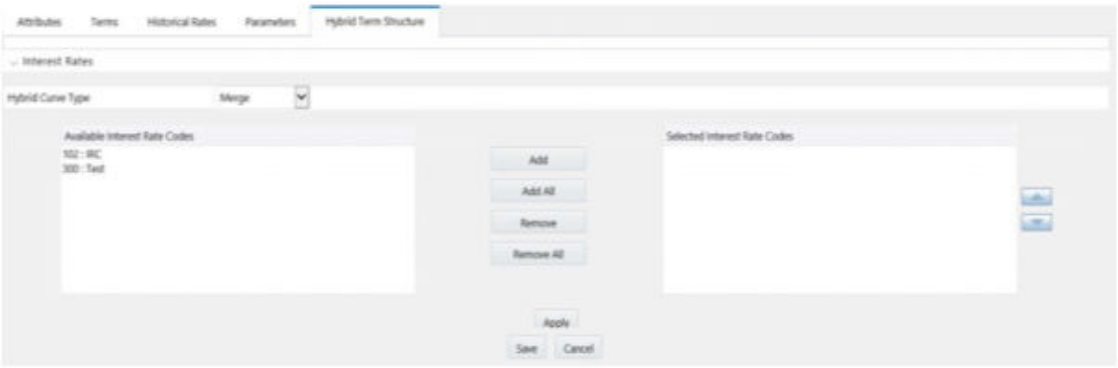
Hybrid Term Structures allows you to specify the following three types of Hybrid yield curves:

- Spread
- Moving Average
- Merge
- Minimum
- Maximum
- Custom Weigthed Average

> **Note**
>
> Minimum and Maximum Hybrid IRC types from the Hybrid IRC type selection are intended for future enhancement and must be ignored in the current release.
> The Parameters and Hybrid Term Structure tabs are disabled if the Volatility Curve is checked.

Hybrid yield curves are built up from either one or more standard yield curves. When you add, modify, or delete any historical rate data from a standard yield curve, the data associated with any related hybrid yield curve must be updated. After defining, the Hybrid yield curves can be used like any other interest rate curve in the system. You can reference these curves within the OFSAA application business rules that allow the selection of an Interest Rate Code.

Figure 9-9 Hybrid Curve tab on Interest Rate Code window



Copyright © 2011, 2026, Oracle and/or its affiliates.

Hybrid Curve Type: :

- **Spread**: A Spread hybrid yield curve is defined as the difference between two standard yield curves. The Spread type of hybrid yield curve is useful in establishing liquidity risk or basis risk yield curves.
- **Moving Average**: Moving average hybrid yield curves represent moving average data of a single underlying standard yield curve. These curves are used in Funds Transfer Pricing.
- **Merge**: Merge hybrid yield curves represent a blending of two or more underlying yield curves. In constructing a Merge type of hybrid yield curve, specify the percentage weighting applied to each of the underlying standard hybrid yield curves.
- **Minimum**: The Minimum hybrid yield curve is defined as the minimum between two standard yield curves.
- **Maximum**: The Maximum hybrid yield curve is defined as the maximum between two standard yield curves.
- **Custom Weighted Average**: Allows to accept weights as input. Here, Effective dates are listed for the term point with maximum historical period over which average need to be calculated. For term points where historical dates are not applicable, you cannot enter weights. The default weight values are zero. Following formula is used to calculate Hybrid Curve Rate for each effective date in historical period = (Weight * Rate) / Total Weight. For more information, see Defining Custom Weighted Average section.
- Define a Hybrid Curve
  To define a hybrid curve, follow these steps:
- Generate Historical Rates
  After a hybrid curve is defined, generate the Historical Rates as far back as the rate source curves allow. The Generate Frequency determines the frequency of the historical rates populated with the Generate function. If you select the Generate Frequency as monthly, it generates month-end values only. If you select daily, it generates the maximum number of historical values.
- Automate (Schedule) the Generate Rates Action in Hybrid IRCs
  The generation of the Hybrid IRC rates is available within the Hybrid IRC of Rate Management UI. As a manual process, you can additionally schedule a Run thorough an ICC Batch process or a Simplified Batch process. If you are running Generate Rates through the UI, then provide the following three parameters:

9.2.2.3.1 Define a Hybrid Curve

To define a hybrid curve, follow these steps:

1. Select the **Source Type** as **Hybrid** in the **Hybrid Term Structure** tab, and then select the **Hybrid Curve Type** (Spread, Moving Average, or Merge).
2. Select the Interest Rate Codes for the hybrid type and click **Apply**.

9.2.2.3.2 Generate Historical Rates

After a hybrid curve is defined, generate the Historical Rates as far back as the rate source curves allow. The Generate Frequency determines the frequency of the historical rates populated with the Generate function. If you select the Generate Frequency as monthly, it generates month-end values only. If you select daily, it generates the maximum number of historical values.

To generate the rates, follow these steps:

1. Select the interpolation method as **Linear**, **Cubic**, or **Quartic**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Select the **Generate Frequency** (Daily, Weekly, Bi-Weekly, or Monthly) and enter the **Specific Date Range** (From Date and To Date).
3. Click **Generate**. The rates will be populated and you will be directed to the Historical Rates tab to view the results.
   If **Hybrid Curve Type** is selected as **Moving Average**, then select the **Moving Average Term** in Days, Months, or Years.

9.2.2.3.3 Automate (Schedule) the Generate Rates Action in Hybrid IRCs

The generation of the Hybrid IRC rates is available within the Hybrid IRC of Rate Management UI. As a manual process, you can additionally schedule a Run thorough an ICC Batch process or a Simplified Batch process. If you are running Generate Rates through the UI, then provide the following three parameters:

- **Interest Rate Code**: For example, you are in edit mode for a particular Hybrid IRC, execute the Generate Rates option. So, the IRC is implied.
- **FROM DATE**: this is the start date of the generate rates process.
- **TO DATE**: this is the date to which hybrid rates will be computed.

When generating Hybrid Rates through a batch (ICC or Simplified), specify ALL in the Parameter list to generate rates for all Hybrid IRCs. Or, a list of IRCs can be specified with comma-separated values within quotes.

'IRC_Code1, IRC_Code2, IRC_Code3' and so on.

> **Note**
>
> Use ALL in the Parameter list for all IRCs.

The **TO DATE** is determined based on the Effective Date specified for the Batch and the **FROM DATE** for each IRC is determined by referring to the last (maximum) effective date in the current historical rates table +1 day.

There is one exception to the calculated FROM DATE. In cases where a new Hybrid IRC is generating rates for the first time, the minimum Effective Date from the Parent IRC is used as the FROM DATE.

> **Note**
>
> Hybrid Rates generated by the Generate Rates procedure are written to the FSI_IRC_RATE_HIST table. Outputs from the procedure can be verified within the Interest Rates UI or by querying this table directly.

Dependent Tables: The following list describes the details of the Dependent Tables.

- **FSI_IRCS**: Holds header information of all Interest Rate Codes.
- **FSI_IRC_HYBRID_STRUCT_WEIGHT**: Holds the Hybrid IRCs' - Interest Rate Term, Interest Rate Term Mult, and Rate Weightage. It is the child Table of FSI_IRCS.
- **FSI_IRC_RATE_HIST**: This Output table holds calculated rates for the Hybrid IRCs, as well as all historical rates for non-hybrid IRCs input through the Interest Rates UI or loaded directly using the interest rates data loader program.
- **FSI_MESSAGE_LOG**: Holds error log information.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **DATE_TASK_MASTER**: Located in the Config Schema. This table must have a seeded date for IRC_HYBRID_SCHEDULER DT.
- **DATE_TASK_STEP_PRECEDENCE**: Similar to DATE_TASK_MASTER, this table must also have an entry for IRC_HYBRID_SCHEDULER DT.

Prerequisites to Running Hybrid IRC Batches:

- The function FN_FSI_IRC_HYBRID_SCHEDULER must be in Compiled status in the Atomic schema. Sometimes, the functions created in the Database can be in an invalid state due to some issues (by the installer), so use the Compile option in the Database to get them in a valid state.
- You can create the ICC batch using the IRC_HYBRID_SCHEDULER rule name, which is seeded with the installation.
- Hybrid IRC rates are calculated based on their underlying standard IRC rates, which are also stored in the FSI_IRC_RATE_HIST table. Historical Rates must already exist for all parent and dependent IRCs for the relevant effective date range.

> **Note**
>
> Seeded data related to Batch information must be present in the DATE_TASK_MASTER and DATE_TASK_STEP_PRECEDENCE tables in the Config schema.

9.2.3 Batch Creation

There are two methods for creating and running the batch processes, ICC Batch and Simplified Batch. The following section describes how to set up and run the Hybrid Rate Generation using both approaches. For more information, see the OFS Analytical Applications Infrastructure User Guide.

Running Hybrid Rate Generation using ICC Batch:

1. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Maintenance**. Click **Create Batch**.
2. Enter the following details in the Batch Maintenance window:
   - Component = "Transform Data"
   - Rule Name = "IRC_HYBRID_SCHEDULER"
   - Parameter List = 'IRC_Code'

   Use ALL in the Parameter list to use all IRCs.
3. Save the batch.
4. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Execution**.
5. Search the **Batch** created under **Batch Maintenance** and select it.
6. Enter the **Information Date** for the batch.
7. Click **Execute Batch**.

   To schedule a future Hybrid Rate Generation Batch Process, or to schedule the Hybrid Rate Generation Batch Process on a recurring basis, do the following:

Copyright © 2011, 2026, Oracle and/or its affiliates.

8. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Scheduler**.
9. Search the **Batch** to schedule and select it.
10. Click **New Schedule** or **Existing Schedule**:

    An Existing Schedule can be selected only if there are existing scheduled batches to view.
11. If New Schedule is selected, the New Schedule details appear. Enter the Scheduled Task Name. To schedule the Hybrid Rate Generation on a recurring basis, select Daily, Weekly, Monthly, or Adhoc.
12. In the Schedule Task pane, enter **Start Date** and **End Date** in the Date field.
13. In the Run Time field, enter the time for the next validation to be run.
14. Click Save to set the schedule as specified or Cancel to drop your changes.

    Any error messages or warnings generated during the Hybrid Rate Generation process that is displayed in the View Log.

    For more information, see the OFS Analytical Applications Infrastructure User Guide.

- Running Hybrid Rate Generation using Simplified Batch
- Viewing Log Messages
  Any error messages or warnings generated during the rate generation batch are displayed in the Log Information window.

9.2.3.1 Running Hybrid Rate Generation using Simplified Batch

To run Hybrid Rate Generation using Simplified Batch, follow these steps:

1. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**.

   (Optional) <Enter a step example.>
2. Click **Add**.
3. From the **Task Details** pane, click **Select Task**. In the **Task Selection** window, select **Task Type** as **Transform Data**. Click **Search**. In the **Task Selector**, select the **IRC_HYBRID_SCHEDULER** and click **Ok**.
4. Enter the Parameters and click **Save**.
5. In the **Simplified Batch Summary** page, search the Hybrid Rate Generation batch, and select it. Click the **Run** icon.
6. Click **Yes** to confirm you want to continue, enter a date (this will be your TO DATE), and click **Ok** to continue. A confirmation message is displayed.
7. Click **Ok**. When the batch is complete, navigate to Operations and select View Log to view the processing log.

   Any error messages or warnings are accessible from the View Log window.

   Simplified Batch does not provide access to logs for Transform Data tasks.

   For more information, see the OFS Analytical Applications Infrastructure User Guide.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.2.3.2 Viewing Log Messages

Any error messages or warnings generated during the rate generation batch are displayed in the Log Information window.

To access this window, follow these steps:

1. Navigate to the Operations menu and select View Log.
2. Enter search criteria as Data Transformation and related Batch ID.
3. Select the Task ID to view the log information.

   You can additionally query the FSI_MESSAGE_LOG table directly to view the error log details.

   For more information, see the OFS Analytical Applications Infrastructure User Guide.

9.3 Currency

Financial institutions transact business in more than one currency. Transacting business in multiple currencies demands functional capabilities for multi-currency accounting and currency rate management.

OFSAA Rate Management's Currency module supports the definitions and maintenance of currencies. Currency definitions are fundamental to the definition of both interest rate yield curves and currency exchange rates. A key attribute of every yield curve is the currency with which it is associated, and currency exchange rates can only be established between defined currencies. OFSAA Rate Management provides a comprehensive list of ISO-defined currencies; you can also define and add your user-defined currencies.

To view the Interest Rate Code Summary page, navigate to **Common Object Maintenance**, select **Rate Management**, and then select **Currency**. This page displays a comprehensive list of more than 170 seeded ISO currency codes.

- Searching for Currency
  A Search pane is provided to search for currencies by Name, Currency (by ISO currency code), Status, or Reporting Currency.
- Editing Currencies
  To edit a currency, select a currency, click Edit. You cannot modify any currency Code value. You can modify the Currency Name, Reference Interest Rate Code, Reporting Currency value (Yes or No), and the currency's Status (Active or Inactive).
- Adding Currencies
  To add a currency, follow these steps:
- Deleting Currencies
  Select one or more currencies and then click **Delete**. You cannot delete any currencies that are utilized elsewhere in the system.

9.3.1 Searching for Currency

A Search pane is provided to search for currencies by Name, Currency (by ISO currency code), Status, or Reporting Currency.

To search the currency, follow these steps:

1. Enter the search criteria and click **Search**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Click the **Reset** to remove the search criteria you have specified and refresh the window.
3. The search results are displayed in a tabular format containing all of the currencies that meet to search criteria. The Currency summary pane offers several icons as follows to perform different functions when a currency is selected.
   - **Add**: Click **Add** to begin the process of adding a new currency. The **Add** icon is disabled if any rows in the pane are selected.
   - **Edit**: Select a single row out of the pane to enable the **Edit**. Click the **Edit** icon to modify an existing currency. The **Edit** icon is only enabled when a single currency has been selected.
     You can control the number of rows to display on the page by selecting the Pagination Options icon from the action bar.
   - **Delete**: Select one or more rows out of the pane to enable the **Delete** icon. Click the **Delete** icon to delete the currency or currencies you have selected.
   - **Check Dependencies**: Select a currency and then click the **Check Dependencies** icon to generate a report on all rules that utilize your selected currency.
     The Check Dependencies icon is only enabled when a single currency has been selected.
4. The **Currencies** window contains all of the currencies that meet the search criteria. The Currencies Summary page has several icons to perform different functions when a currency is selected.
   To select a currency, select the check box in the first column of the pane. More than one currency can be selected at a time but this will cause some of the icons to become disabled. Select a check box a second time deselects the currency.

   Select or deselect all of the currencies in the Summary page by selecting the check box in the upper left-hand corner of the Summary page directly to the left of the Code column header.

   The following list provides the currency details based on the search criteria:
   - **Code**: For seeded currencies, these are ISO Currency Codes. For user-defined currencies, these can be any pure character string (no numbers) up to a length of 3 characters.
   - **Currency Name**: For seeded currencies, these are ISO Currency Codes. For user-defined currencies, these can be any string up to a length of 40 characters.
   - **Reference Interest Rate Code**: Reference Interest Rate Code is the Interest Rate Code with which currency is associated with forecasting purposes. Define multiple yield curves each of which has the same Reference Currency, but a currency can only have one Reference Interest Rate Code.
   - **Reporting Currency**: A reporting currency is an active currency to which balances in other currencies can be consolidated to facilitate reporting. Balances in reporting currencies can be, in turn, consolidated to the functional currency. For example, an American multinational bank might consolidate its holdings in Asian currencies to the Japanese yen (Reporting Currency) and its balances in European currencies to the Euro (Reporting Currency) after which it might consolidate these reporting currencies to the U.S. dollar (Functional Currency).
   - **Status**: The status of any currency can be either Active or Inactive. You must Activate a currency before doing the following:
     Define that currency as a Reference Currency for a yield curve (For more information, see the **Reference Currency** section in the Interest Rates Details window.)

     Enter Exchange Rate data for a currency (For more information, see the Currency Rates section.)

Copyright © 2011, 2026, Oracle and/or its affiliates.

Define Forecast Rates for that currency within OFS ALM (For more information, see the Forecast Rates section).

9.3.2 Editing Currencies

To edit a currency, select a currency, click Edit. You cannot modify any currency Code value. You can modify the Currency Name, Reference Interest Rate Code, Reporting Currency value (Yes or No), and the currency's Status (Active or Inactive).

You cannot inactivate any currency that is:

- Defined as the Reference Currency for any yield curve
- Associated with any exchange rate data
- Utilized within any Forecast Rates rule within OFSAA Asset/Liability Management

After completing the edits, click **Save**.

9.3.3 Adding Currencies

To add a currency, follow these steps:

The first row in the Currencies window is an empty row that is ready for you to edit as a new currency. Select this empty row, click **Edit**, and generate the new currency.

or

Generate a new currency by clicking **Add**. This generates another empty row at the top of the Currencies window to edit purpose.

9.3.4 Deleting Currencies

Select one or more currencies and then click **Delete**. You cannot delete any currencies that are utilized elsewhere in the system.

9.4 Currency Rates

OFSAA Rate Management's Currency Rates module uses the currencies defined and activated in the Currency module to support the creation and maintenance of historical exchange rates. Currency exchange rates are utilized in:

- OFS Funds Transfer Pricing (OFS FTP) Ledger Migration processes
- OFS Asset Liability Management (OFS ALM) Currency Consolidation process (see the OFS Asset Liability Management User Guide)
- OFS Profitability Management multi-currency allocations.
- OFS Hedge Management and IFRS Valuations (see the Oracle Hedge Management and IFRS Valuations User Guide)

To view the Currency Rate, navigate to Common Object Maintenance, select Rate Management, and then select Currency Rates. In this window, you can manage historical exchange rates between currencies.

For ALM, FTP, and PFT, From Currency defaults to the Initial Currency selection from the Assumption Management defaults in the active Application Preferences window. You can select another From Currency from the drop-down list that displays all Active currencies.

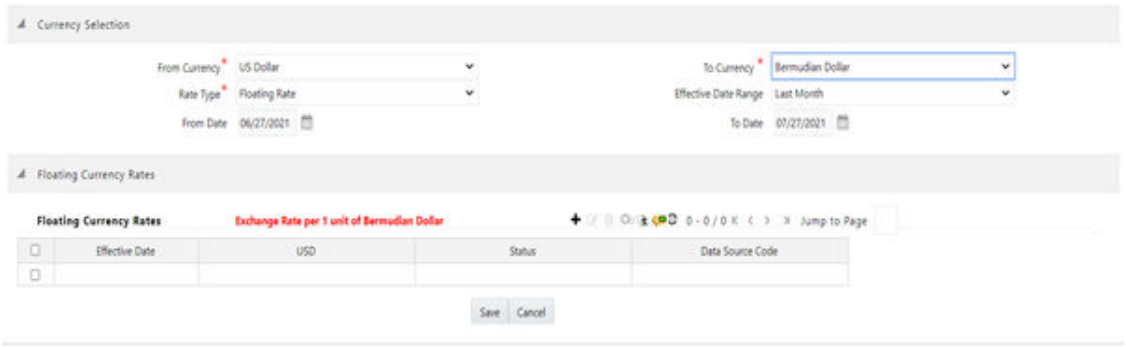
Copyright © 2011, 2026, Oracle and/or its affiliates.

- Preparing to Work with Exchange Rate Data
  To view, edit, or delete the exchange rate data, you must enter the To Currency value. With the From Currency, the To Currency drop-down list displays only Active currencies.
- Adding Exchange Rate Data
  After you have specified a value for To Currency, the Floating Currency Rates section appears. To define a Fixed Rate relationship, select the Fixed Rate Type and replace the Floating Currency Rates with the Fixed Currency Rates.
- Adding Multiple Exchange Rates
  Click Add to add additional blank rows to enter the additional Effective Dates and Exchange Rates. After adding the multiple new exchange rates, click **Save**.
- Editing Exchange Rate Data
  Select the check box on the left-hand side of any row to enable the Edit icon. After clicking Edit, the row becomes active to edit the Effective Date and (or) the Exchange Rate. Click **Save** to save the changes.
- Viewing Exchange Rate Data
  By default, both the Floating Currency Rates pane and the Fixed Currency Rates pane display the most recent month of historical exchange rate data. You can control the amount of data displayed by selecting a different value from the Effective Date Range drop-down list in the Currency Selection window.
- Deleting Exchange Rate Data
  Select one or more check boxes on the left-hand side of any row to enable the Delete icon. After clicking **Delete**, a confirmation message is displayed. Click **Ok**.
- Data Loader
  The Data Loader icon executes a function to import historical Currency rates for all defined Currencies. For more information on setting up the automated process, see the OFS Data Model Utilities User Guide.
- Excel Compatibility
  This functionality is used for adding or editing historical exchange rates from an Excel sheet. You can right-click on any cell in the table and perform the following actions:

9.4.1 Preparing to Work with Exchange Rate Data

To view, edit, or delete the exchange rate data, you must enter the To Currency value. With the From Currency, the To Currency drop-down list displays only Active currencies.

Figure 9-10 Currency Rates Summary page



After selecting a To Currency value, select a value for Rate Type as Floating Rate or Fixed Rate (the default selection is Floating Rate). After selecting a To Currency value, a second

Copyright © 2011, 2026, Oracle and/or its affiliates.

pane is displayed as Floating Currency Rates (if you have chosen a Rate Type of Floating Rate) or Fixed Currency Rates (if you have chosen a Rate Type of Fixed Rate).

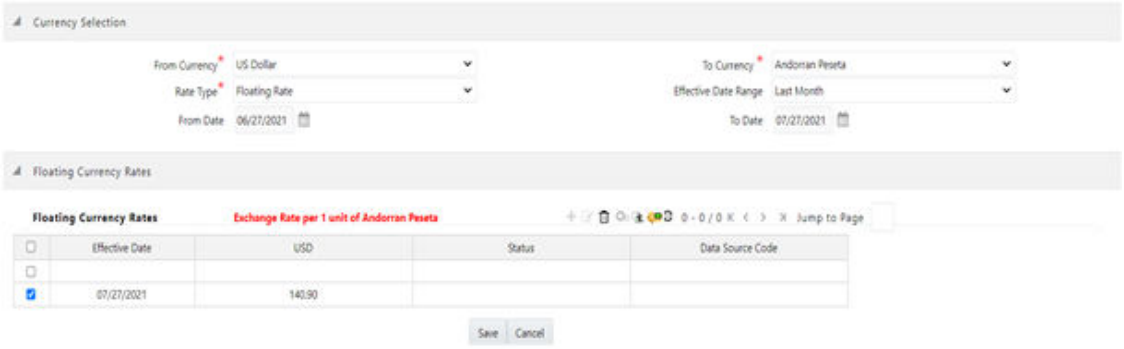
- **Floating Rates**: Floating exchange rates, such as those between the US Dollar (USD), the Pound Sterling (GBP), the Japanese Yen (JPY), and the Euro (EUR), are market-driven and can change from day-to-day, hour-to-hour, or minute-to-minute.
- **Fixed Rates**: Some countries, especially smaller countries or countries that have experienced significant inflation in the recent past, can wish to "peg" their currency to a larger, more stable currency such as the US Dollar, Japanese Yen, or Euro.

9.4.2 Adding Exchange Rate Data

After you have specified a value for To Currency, the Floating Currency Rates section appears. To define a Fixed Rate relationship, select the Fixed Rate Type and replace the Floating Currency Rates with the Fixed Currency Rates.

Both Currency Rates panes initially display a single blank row followed by the most recent month's exchange rate data (if any such exchange rate data already exists). To enter a single new exchange rate data point, enter data into the blank row, and click **Save**.

Figure 9-11 Defining a Currency Rate



The following list describes the columns in the Currency Rate window:

- • Effective Date: Directly enter a date or select the Calendar ( ) icon to choose an effective date for your new exchange rate data point.
  Rate Management stores the historical exchange rate data. You cannot enter exchange rate data for dates greater than the current date. For more information regarding rate forecasts and the relationship between historical exchange rates for forecasted exchange rates, see the Forecast Rate Scenarios. If you have gaps in the historical exchange rate data, any OFS Analytical Application that needs to perform a rate translation function will fall back to the most recent date for which exchange rate data exists. For example, if an OFS Analytical Application needs to translate a rate from USD to EUR for February 22, 2010, and the latest available USD to EUR rate data in the Rate Management database is February 11, 2010, then the application will utilize the exchange rate for February 11, 2010.
- **Status**: Status is a read-only display that is updated after the Currency Rates Validation has been run.
- **Data Source Code**: The Data Source Code is displayed read-only and indicates whether the rates were input through the UI or the data loader.
- **Currency Exchange Rate**: For both Floating Rates and Fixed Rates, units of the From Currency are converted to one unit of the To Currency. See Table 12 for example.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 9-3 Example of one Currency to another Currency Conversion

From Currency	To Currency	Approximate Rate
USD - US Dollar	GBP-Pound Sterling	1.50
USD - US Dollar	EUR - Euro	1.36
USD - US Dollar	JPY - Japanese Yen	0.01105

9.4.3 Adding Multiple Exchange Rates

Click Add to add additional blank rows to enter the additional Effective Dates and Exchange Rates. After adding the multiple new exchange rates, click **Save**.

9.4.4 Editing Exchange Rate Data

Select the check box on the left-hand side of any row to enable the Edit icon. After clicking Edit, the row becomes active to edit the Effective Date and (or) the Exchange Rate. Click **Save** to save the changes.

9.4.5 Viewing Exchange Rate Data

By default, both the Floating Currency Rates pane and the Fixed Currency Rates pane display the most recent month of historical exchange rate data. You can control the amount of data displayed by selecting a different value from the Effective Date Range drop-down list in the Currency Selection window.

Click the **View** icon to a specific range of effective dates by modifying the From Date, To Date, or both dates within the Currency Rates window.

9.4.6 Deleting Exchange Rate Data

Select one or more check boxes on the left-hand side of any row to enable the Delete icon. After clicking **Delete**, a confirmation message is displayed. Click **Ok**.

9.4.7 Data Loader

The Data Loader icon executes a function to import historical Currency rates for all defined Currencies. For more information on setting up the automated process, see the OFS Data Model Utilities User Guide.

To execute a data load, click the **Data Loader** icon. A warning message is displayed *Upload all available Currency Rates?*

Click the **Data Loader** icon to execute all the currency rates.

9.4.8 Excel Compatibility

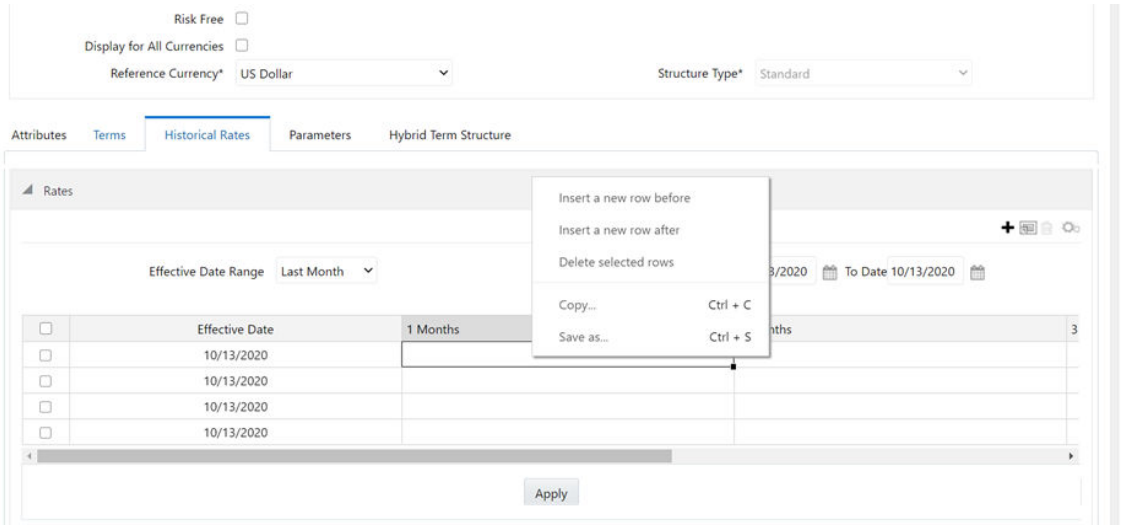
This functionality is used for adding or editing historical exchange rates from an Excel sheet. You can right-click on any cell in the table and perform the following actions:

- Delete or Insert multiple rows.
- Apply a Formula on a particular column.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Export the data to excel using the **Save As** option.
- You can copy data from excel and paste the same using standard copy, paste shortcuts **Ctrl+C** and **Ctrl+V**.

Figure 9-12 Historical Rates Tab



9.5 Currency Exchange Rate Validation

Exchange Rate Validation has the following features:

- Movement of historical exchange rates to the Currency Direct Access table.
- Calculation of inverse exchange rates for reporting currencies.
- Calculation of triangulated exchange rates where possible.

> **Note**
>
> This feature is equivalent to the Rate Validation capability in OFSA 4.5 Rate Manager.

- Features of Exchange Rate Validation
  The goal of exchange rate validation is to ensure that exchange rates from all active currencies to all reporting currencies are available for processing in the OFSAA applications. Some of these rates can come from the validated direct input, others are calculated based on relationships with other rates. To support triangulation, all fixed exchange rates are available for all currencies that make up an exchange that needs to be triangulated. Also, a direct exchange rate between each Child currency and each reporting currency is calculated and supplied to support quick access to exchange rates. If a Child currency is a Reporting Currency, then Exchange Rates are calculated for all currencies having an exchange relationship with the Parent currency.
- Currency Rate Validation
  Load currency rates through the Rate Management or using the Historical Rate Data Loader. The initial status is Not Yet Validated.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Validating Exchange Rate Relationships
  You must run the exchange rate validation process after adding or modifying exchange rate data. Run the process immediately or schedule one or more to be run in the future.
- Running an Exchange Rate Validation
  You can run a validation immediately or schedule one or more for later. The validation status is displayed in the Currency Rates window of the Rate Management.
- Scheduling One or More Validations
- Viewing the Messages
  Any error messages or warnings generated during the exchange rate validation process are displayed in the Log Information window.

9.5.1 Features of Exchange Rate Validation

The goal of exchange rate validation is to ensure that exchange rates from all active currencies to all reporting currencies are available for processing in the OFSAA applications. Some of these rates can come from the validated direct input, others are calculated based on relationships with other rates. To support triangulation, all fixed exchange rates are available for all currencies that make up an exchange that needs to be triangulated. Also, a direct exchange rate between each Child currency and each reporting currency is calculated and supplied to support quick access to exchange rates. If a Child currency is a Reporting Currency, then Exchange Rates are calculated for all currencies having an exchange relationship with the Parent currency.

9.5.2 Currency Rate Validation

Load currency rates through the Rate Management or using the Historical Rate Data Loader. The initial status is Not Yet Validated.

1. Execute the Currency Rate Validation engine.
2. Navigate to **Operations**, select **Batch Maintenance**, and then click **Create Batch**.
   Component = "Transform Data"

   Rule Name = "Rate_Validation"

   Parameter List (Required) = FROM_DATE and TO_DATE, for example, '20110101', '20110131'

   After completion of the validation process, the status in the Currency Rates window changes to Valid or Invalid (in case there is an issue).
3. View both input and calculated results in the FSI_EXCHNG_RATE_DIRECT_ACCESS table. The OFSAA processing engine reads this table for sourcing historical exchange rates.

> **Note**
>
> Executing Rate Validation is a required step when a multi-currency setup is enabled. All of the OFSAA processing engines require exchange rates to be validated through this process to convert results from base currency to a selected reporting currency. If exchange rate validation is not run or required exchange rates are not available, the engines will use default exchange rates = 1.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.5.3 Validating Exchange Rate Relationships

You must run the exchange rate validation process after adding or modifying exchange rate data. Run the process immediately or schedule one or more to be run in the future.

Each exchange rate has one of the following statuses:

- **Not Yet Validated**: The exchange rate has been input or loaded but not yet validated.
- **Valid**: The exchange rate has been validated.
- **Invalid**: The exchange rate has violated one or more acceptance rules.

Only exchange rates in Valid status are available for processing and they are not subject to future validation unless you edit them. The Rate Validation status is displayed in the Currency Rates window of the Rate Management.

Exchange Rate Validation Criteria: In the rate validation process, all exchange rate relationships in the database are examined for compliance with the following criteria. Error messages and warnings are displayed if one or more criteria are not met. For more information, see the Viewing Log Messages.

If a currency is defined as a Child in a fixed exchange relationship then it must not be in any floating (standard) exchange rate relationship at the same time. Consequently, all floating exchange rates to or from the Child currency must be defined through the Parent currency. If this criterion is not met then the following message is displayed: Invalid fixed relationship-Child currency exists in a standard exchange rate within the same period.

- A Child currency within a fixed relationship must not be a Child currency in any other fixed relationship during the same period. If this criterion is not met then the following message is displayed: Invalid fixed relationship-Child currency already exists in a fixed relationship for the same period.
- A circular relationship must not exist. In other words, a Child currency cannot link back to its Parent in any other fixed-rate relationship within the same period. If it does, then the following message is displayed: Invalid fixed relationship creates a circular relationship with other fixed exchange rates.
- Regarding new floating (standard) exchange rates, From and To currencies must not exist as Child currencies within any fixed exchange rate relationships. If this criterion is not met then the following message is displayed: From or To or Both currency or currencies in the new exchange rate already exist in a fixed relationship for the same period.
- If any exchange rate is equal to 0, then a warning message is displayed. 0 is a valid value. You can use it, for example, to designate an exchange rate with a currency of a country that no longer exists.

If two exchange rate relationships fail to meet these criteria then both of them will be labeled Invalid. (Exception, if one of the relationships is already in Valid status, then the other one will be labeled Invalid.) For example, if a currency is defined as a Child in a fixed rate relationship and is also defined as being in a floating relationship at the same time, then both fixed and floating rates for that currency will be labeled Invalid.

If there are both direct and inverse floating exchange rates defined for any two currencies (in other words, one currency is both a To and a From currency in relation to the other), then both relationships will be marked Valid.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.5.4 Running an Exchange Rate Validation

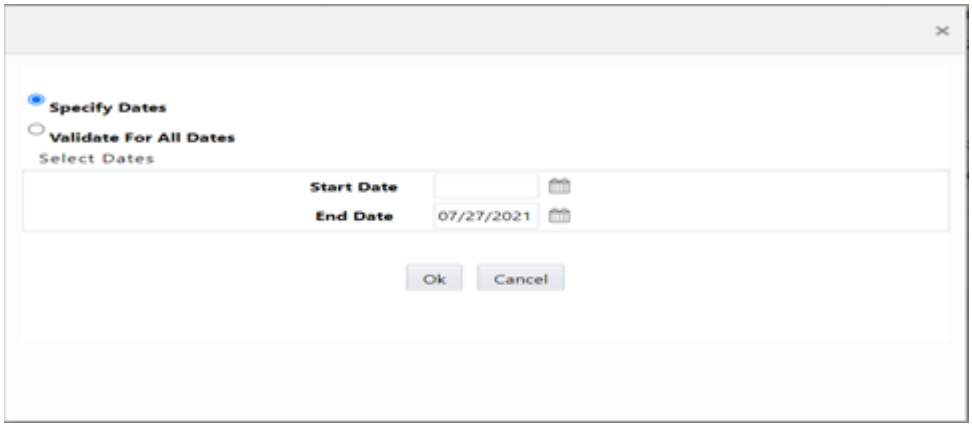
You can run a validation immediately or schedule one or more for later. The validation status is displayed in the Currency Rates window of the Rate Management.

- Running a Validation Immediately
  Execute the exchange rate validation using the Currency Rates Validation option.

9.5.4.1 Running a Validation Immediately

Execute the exchange rate validation using the Currency Rates Validation option.

Figure 9-13 Currency Rate Validation window



To execute the exchange rate validation, follow these steps:

1. Select a check box on the left-hand side of any row to enable the **Currency Rates Validation** icon.
2. Click **Currency Rates Validation**.
3. To execute exchange rate validation from the Currency Rates window, the following two options are available:
   - **Specify Dates**: After selecting this option, a Select Dates pane is displayed to enter or verify the FROM_DATE and TO_DATE parameters. These dates will be passed to the batch for execution.
   - **FROM_DATE**: This defaults to the date of last rate validation.
   - **TO_DATE**: This defaults to the current date.
   - **Validate For All Dates**: Select this option to validate all the rates irrespective of dates.

> **Note**
>
> This option will replace all of the validated exchange rate history and can be a time-consuming process depending on the amount of history available to be processed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

You can execute rate validation using a Simplified Batch or the ICC Batch window. You can also launch it from the Currency Rates UI (Currency Rates Validation toolbar).

To run the validation using the Simplified Batch, follow these steps:

1. Click **Common Object Maintenance**, select **Operations**, and then select **Simplified Batch**, and then click **Add**.
2. From the **Task Details** pane, click **Select Task**. In the **Task Selection** window, choose **Task Type** as **Transform Data**. Click **Search**. In the **Task Selector** window, select **Rate_Validation** and click **Ok**.
3. Back in the **Simplified Batch Definition** window, enter the optional parameters to specify the From and To dates, using the format YYYYMMDD, YYYYMMDD.
   From_Date and To_Date must not be the same value. Set From_Date equal to the last rate validation date and To_Date to the current date. This will ensure that the Effective To Date for the prior record is set correctly.
4. Click **Save**.
5. In the **Simplified Batch Summary** page, search for and select your batch, then click **Run**.
6. Click **Yes** to confirm you want to continue, enter a Date and click **Ok** to continue. You will receive a message that it has been successfully launched.
7. Click **Ok**.
   When the batch is complete, optionally navigate to Operations and select View Log to view the processing log.

   Any error messages or warnings are displayed in View Log. For more information, see Viewing the Messages.

   Simplified Batch does not yet provide access to logs for Transform Data tasks.

For more information, see the OFS Analytical Applications Infrastructure User Guide.

To run the validation using the ICC Batch Framework, follow these steps:

1. Navigate to **Operations**, select **Batch Maintenance**, and then click **Add** to create a new batch.
2. Search for the above batch, and select it.
3. In the **Task Details** pane (tool-bar), click **Add**. In the **Task Selection** window, choose the **Task Type** as **Transform Data**. Click **Search**. In the Task Selector, select the **Rate_Validation** task and click **Ok**.
4. Back in the **Task Definition** window, select the **Rule Name** as **Rate_Validation** and enter the optional parameters to specify the From and To dates, using the format YYYYMMDD, YYYYMMDD.
   From_Date and **To_Date** must not be the same value. Set **From_Date** equal to the last rate validation date and **To_Date** to the current date. This will ensure that the Effective To Date for the prior record is set correctly.
5. Click **Save**.
6. Navigate to **Operations**, and select **Batch Execution**. Search your batch and select it. Enter Information Date and click **Execute Batch**.
7. Click **Ok**. A message is displayed stating Batch triggered successfully, also displaying the Batch Run ID.
8. Click **Ok**.
9. To view rates, query the database table FSI_EXCHNG_RATE_DIRECT_ACCESS after the run is complete.

Copyright © 2011, 2026, Oracle and/or its affiliates.

FSI_EXCHNG_RATE_DIRECT_ACCESS: This output table holds the valid currency codes and their exchange rate with respect to the reporting currency for some time say From Date value to optional To Date value. When Exchange Rate Validation runs, if data already exists for the selected date range, the Rate Validation package will overwrite the existing values. If no dates are given, the Rate Validation package truncates this table and re-loads data for the entire historical date range. The following describes the structure of the FSI_EXCHNG_RATE_DIRECT_ACCESS table:

Table 9-4 Column details of FSI_EXCHNG_RATE_DIRECT_ACCESS table

Column Name	Nullable	Data Type
FROM_CURRENCY_CD	Not Null	varchar2(15)
TO_CURRENCY_CD	Not Null	varchar2(15)
EFFECTIVE_FROM_DATE	Not Null	Date
EFFECTIVE_TO_DATE	Not Null	Date
EXCHANGE_RATE	Not Null	Number(15,9)
EXCHANGE_RATE_CONVERT_TYPE_CD	Not Null	Number(5)
EXCHANGE_RATE_CONV_FORMULA	Not Null	Number(15)

10. When the batch is complete, navigate to Operations and select View Log to view the processing log. Any error messages or warnings are displayed in View Log. For more information, see Viewing the Messages.
For more information, see the OFS Analytical Applications Infrastructure User Guide.

9.5.5 Scheduling One or More Validations

To schedule a future validation, or to schedule validations on a recurring basis, follow these steps:

1. Navigate to **Operations** and select **Batch Scheduler**. The **Batch Scheduler** window appears.
2. In the **Batch Name** pane, select the **Scheduled Batch ID**. An unchecked box means that no validation is scheduled to run.
3. Select **New Schedule** or **Existing Schedule**.

   The Existing Schedule can be selected only if there are existing schedule batches to view.

   If New Schedule is selected, the New Schedule pane appears. Enter the Scheduled Task Name.

   To schedule the validation on a recurring basis, select Daily, Weekly, Monthly, or Adhoc.
4. In the **Schedule Task** pane, enter the **Start Date** and **End Date** in the **Date** field.
5. In the **Run Time** field, enter the time for the next validation to be run.
6. Click **Save** to set the schedule as specified or Cancel to drop your changes.

   Any error messages or warnings generated during the validation process are displayed in the View Log. For more information, see Viewing the Messages.

   For more information, see the OFS Analytical Applications Infrastructure User Guide.

Copyright © 2011, 2026, Oracle and/or its affiliates.

9.5.6 Viewing the Messages

Any error messages or warnings generated during the exchange rate validation process are displayed in the Log Information window.

To view the messages, follow these steps:

1. Navigate to **Operations** and select **View Log**.
2. Click the **Task ID**. The Log Information window will display.

These exception messages can also be seen in the FSI_MESSAGE_LOG table with the help of the batch_id which was used during execution.

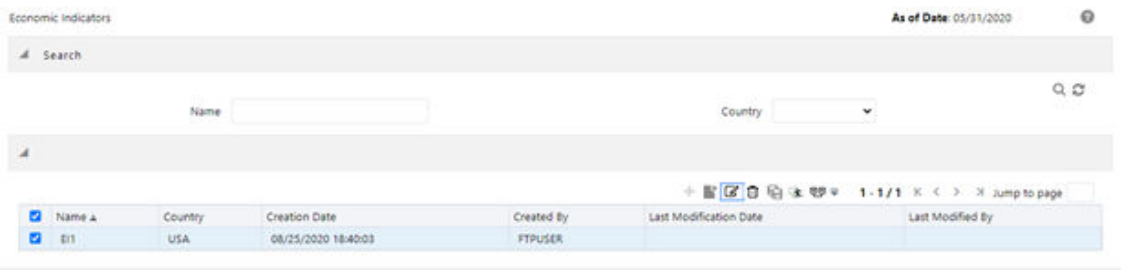
9.6 Economic Indicators Summary Page

An economic indicator is any economic statistic such as the Consumer Price Index (CPI), growth rate of the Gross Domestic Product (GDP), unemployment rate, Purchasing Managers Index, indices of consumer confidence, and so on. Such macroeconomic statistics tell us how well the economy has behaved in the past. Some economic indicators are referred to as lagging indicators while others are classified as leading indicators. Leading indicators can provide insights into the future direction of the economy.

OFSAA Rate Management's Economic Indicators module allows you to define and store such historical indicators. Economic Indicators provide baselines from which OFSAA Asset Liability Management can generate forecasts of future values of economic statistics that can affect new business or other modeling assumptions.

To view the Economic Indicators, navigate to Common Object Maintenance and select the Rate Management, and then select Economic Indicators, an empty window is displayed. After you have defined one or more Economic Indicators, the Economic Indicators Summary page shows all the Economic Indicators that you have previously built.

Figure 9-14 Economic Indicators Summary page



- Searching for Economic Indicator
  A Search pane is provided in which you can search for Economic Indicators by Name or Country. Each Economic Indicators rule is specific to one country.
- Economic Indicators Details
  When you **Add**, **Edit**, or **View** an Economic Indicator, the Economic Indicator Details window is displayed. The **Economic Indicator Detail** window includes an **Economic Indicator Details** pane, the **Economic Indicators - Historical Data** pane, and an **Audit Trail** pane.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Adding an Economic Indicator
  To add an Economic Indicator, follow these steps:

9.6.1 Searching for Economic Indicator

A Search pane is provided in which you can search for Economic Indicators by Name or Country. Each Economic Indicators rule is specific to one country.

To search the Economic Indicator, follow these steps:

1. Enter search criteria and click the **Search** icon.
2. Click the **Reset** icon to remove any Name or Country search criteria and refreshes the window.
   The search results are displayed in a tabular format containing all of the Economic Indicators that meet search criteria. The Economic Indicators Summary page offers several icons to perform different functions when an Economic Indicator is selected.
   - **Add**: Click Add to begin the process of building a new Economic Indicator. The Add icon is disabled if any rows in the pane are selected.
   - **View**: Select a single row out of the pane to enable the View. Click the View icon to view the contents of an Economic Indicator on a read-only basis. The View icon is only enabled when a row is selected.
   - **Edit**: Select a single row out of the pane to enable the Edit. Click the Edit icon allows you to modify a previously saved Economic Indicator. The Edit icon is only enabled when a single row is selected.
   - **Delete**: Select one or more rows out of the pane to enable the Delete. Clicking on the Delete deletes the Economic Indicators you have selected.
   - **Copy**: Select a single row out of the pane to enable the Copy. Click the Copy icon to create a copy of an existing economic indicator. The Copy icon is only enabled when a single economic indicator is selected.
   - **Check Dependencies**: Select an Economic Indicator and then click the Check Dependencies icon to generate a report on all rules that utilize your selected economic indicator.
     The Check Dependencies is only enabled when a single economic indicator is selected.
   - **Data Loader**: The Data Loader option executes a function to import historical economic indices for all defined Economic Indicators. For more information on setting up the automated process, see the OFS Data Model Utilities User Guide.
     To execute a data load, click on the **Data Loader**.

     A warning message will appear Upload all available Economic Indicators? Click Ok, and all historical indices will be loaded.

To select an Economic Indicator, select a check box in the first column of the pane. More than one Economic Indicators can be selected at a time but this will cause some of the icons to become disabled.

Select or deselect all of the Economic Indicators in the Economic Indicators Summary page by selecting the check box in the upper left-hand corner of the Summary page directly to the left of the Name column header.

The following columns categorize each Economic Indicator on the Summary page:

- **Name**: Displays the Economic Indicator's short name. Performing a mouse-over on a row within the pane displays the Economic Indicator's detailed description.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Country**: Displays the Country to which an Economic Indicator applies.
- **Creation Date**: Displays the Date and Time at which an Economic Indicator was created.
- **Created By**: Displays the Name of the user who created an Economic Indicator.
- **Last Modification Date**: Displays the Date and Time at which an Economic Indicator was last modified.
- **Modified By**: Displays the Name of the user who last modified an Economic Indicator.

9.6.2 Economic Indicators Details

When you **Add**, **Edit**, or **View** an Economic Indicator, the Economic Indicator Details window is displayed. The **Economic Indicator Detail** window includes an **Economic Indicator Details** pane, the **Economic Indicators - Historical Data** pane, and an **Audit Trail** pane.

The **Audit Trail** pane is a standard footer pane for every OFSAA rule type. The **Audit Trail** pane displays Created By, Creation Date, Last Modified By, and Modification Date. The **User Comments** tab can be used to add comments to any rule, subject to a maximum of 4000 characters.

9.6.3 Adding an Economic Indicator

To add an Economic Indicator, follow these steps:

1. Click **Add** from the Economic Indicator Summary page.
2. Enter the following information in the Economic Indicator Details window as tabulated.
   - **Name**: The name of your Economic Indicator is how you will subsequently refer to your rule within the OFS Analytical Applications. You cannot rename existing Economic Indicators.
   - **Frequency**: The frequency of your Economic Indicator must match the frequency with which the indicator's data is made public. Unemployment statistics, for example, are normally released on a monthly frequency. Select a frequency from the Frequency drop-down list. Available frequencies are **Weekly**, **Monthly**, **Quarterly**, **Semi-Annually**, and **Annually**.
   - **Value Type**: Select a Value Type from the **Value Type** drop-down list. Available Value Types are Numeric, Percentage, and Amount.
     - **Numeric**: 0-999999
     - **Percentage**: -100 to +100
     - **Amount**: 0-999999
   - **Country**: Select a country to which your Economic Indicator applies from the Country drop-down list. The value set of Countries is drawn from the seeded Country dimension. OFSAA is seeded with over 70 country values, and you can add user-defined countries.
3. Click **Save**. To build out your historical data, enter data within the Economic Indicators - Historical Data pane.
   The Economic Indicators - Historical Data pane displays a single blank row followed by the most recent period's data (if data has previously been stored in the database).

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 9-15 Economic Indicators - Historical Data section on Economic Indicator window

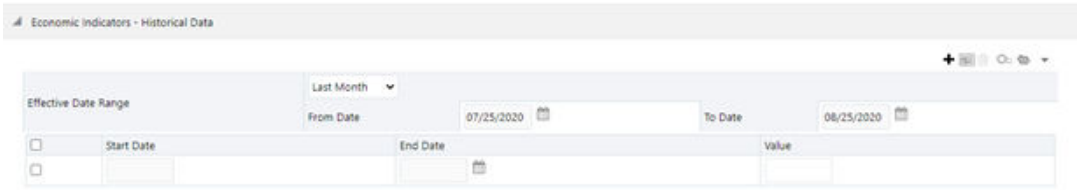


Table 9-5 Economic Indicators - Historical Data section - Fields and Descriptions

Fields	Description
Start Date and End Date	Select the **Calendar** icon immediately adjacent to the End Date to choose an ending date for your Economic Indicator data point. The application will automatically populate the Start Date based on the Economic Indicator's frequency. For example, if your Economic Indicator is an unemployment statistic that has a monthly frequency, select an end date that is the last day of the month that the unemployment rate describes. In this example, the application will automatically populate the Start Date with the first day of the month you have selected.
Value	Enter the value for your Economic Indicator (such as the unemployment rate).
Adding Multiple Data Points	Click **Add** to add additional blank rows into which you can enter additional Economic Indicator data. When you have finished adding data, click Save.
Editing Economic Indicators - Historical Data	Select a single check box on the left-hand side of any row to enable the Edit. Clicking **Edit** to become the row active. You can edit this row and subsequently save the changes.
Viewing Economic Indicators - Historical Data	By default, the Economic Indicators - Historical Data pane displays the most recent month of historical data. You can restrict the amount of data displayed by selecting a different value from the **Effective Date Range** drop-down list. You can also choose to view a specific date range by modifying the **From Date**, **To Date**, or both dates within the **Economic Indicators - Historical Data** pane.
Deleting Economic Indicators - Historical Data	Select one or more check boxes on the left-hand side of any row to enable the **Delete** icon. After clicking Delete, a confirmation message is displayed to delete the selected rows.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 9-5 (Cont.) Economic Indicators - Historical Data section - Fields and Descriptions

Fields	Description
Data Loader	The Data Loader option executes a function to import historical economic indicators for all defined economic indicators. For more information on setting up the automated process, see the OFS Data Model Utilities User Guide. To execute a data load, click the Data Loader. A warning message will appear Upload all available Economic Indicators? Click **Data Loader** to execute all the Economic Indicators.
Excel Import and Export	Use Excel Import or Export functionality to add or edit Historical Economic Indicators.

Copyright © 2011, 2026, Oracle and/or its affiliates.

10

Holiday Calendar

This chapter discusses the procedure to create a Holiday Calendar and generate a list of the weekend and holiday dates. Individual applications may consume the Holiday Calendar events in different ways. For more information, see the application-specific User Guides and the OFS Cash Flow Engine Reference Guide.

A Holiday is a day designated as having special significance for which individuals, a government, or some religious groups have deemed that observance is warranted and thus no business is carried on this day. The Holiday Calendar code can range from 1 to 99999.

The procedure for working with and managing a Holiday Calendar is similar to that of other OFSAA business rules.

Topics:

- Searching for a Holiday Calendar
- Creating a Holiday Calendar
- Executing Holiday Calendar
- Holiday Exceptions
- Searching for a Holiday Calendar
  Search for a Holiday Calendar to perform any of the following tasks:
- Creating a Holiday Calendar
  You create Holiday Calendars to capture holidays for a given date range for any organization. It is possible to create and use multiple Holiday Calendars.
- Excel Import and Export
- Executing Holiday Calendar
  Execute a Holiday Calendar definition to generate calendar dates listing the various types of holidays for a given holiday period.
- Holiday Exceptions
  You can specify exceptions to the holidays.

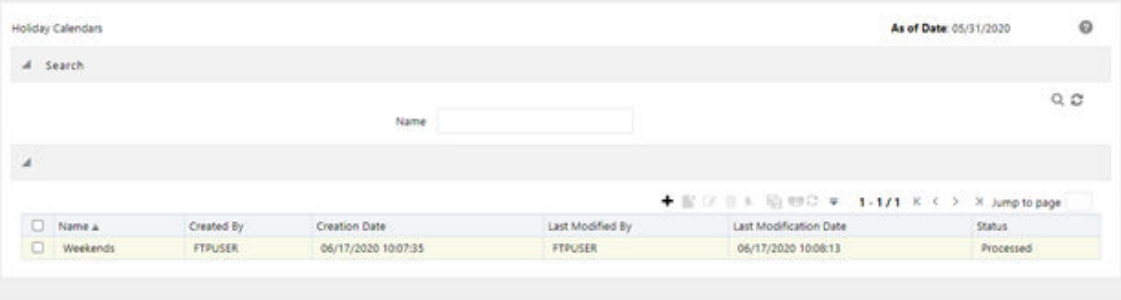
10.1 Searching for a Holiday Calendar

Search for a Holiday Calendar to perform any of the following tasks:

- View
- Edit
- Copy
- Delete
- Check Dependencies
- Refresh

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 10-1 Holiday Calendars



You must have created a Holiday calendar to be able to search for Holiday Calendar rules.

To search for a new Holiday Calendar rule, follow these steps:

1. From the LHS menu, select **Common Object Maintenance**, and then select **Holiday Calendar** to display the **Holiday Calendars** Summary page. This page holds all Holiday Calendars and related functionality. You can navigate to other pages relating to the Holiday Calendar from this page.
2. Enter the name of the Holiday Calendar and click **Search**.
   Only Holiday Calendars that match the search criteria are displayed.

> **Note**
>
> You can control the number of rows to display on the screen by selecting the Pagination icons from the action bar.

10.2 Creating a Holiday Calendar

You create Holiday Calendars to capture holidays for a given date range for any organization. It is possible to create and use multiple Holiday Calendars.

The Holiday Calendar functionality allows you to define two types of holidays: Fixed and Moving.

- A fixed holiday is deemed as a holiday for every year in the holiday period, for that particular day. Example: 25th December – Christmas, is a fixed holiday. For a fixed holiday, the Holiday Calendar procedure will populate entries in the holiday list table for the selected date range.
- A moving holiday is deemed as a holiday only for that particular date and year, and not for every year in the holiday period. All occurrences of a moving holiday must be input manually. Example: 10th April 2020 is a moving holiday for Good Friday.
- A HOLIDAY_TYPE code:
  - 0: Fixed type
  - 1: Moving type
  - 2: Weekend

The Holiday Calendar procedure will also ensure that holiday and weekend entries are not duplicated. For example, if weekends are defined as Saturday/Sunday and Christmas falls on a weekend day, there will be only one entry in the FSI_HOLIDAY_LIST table. The

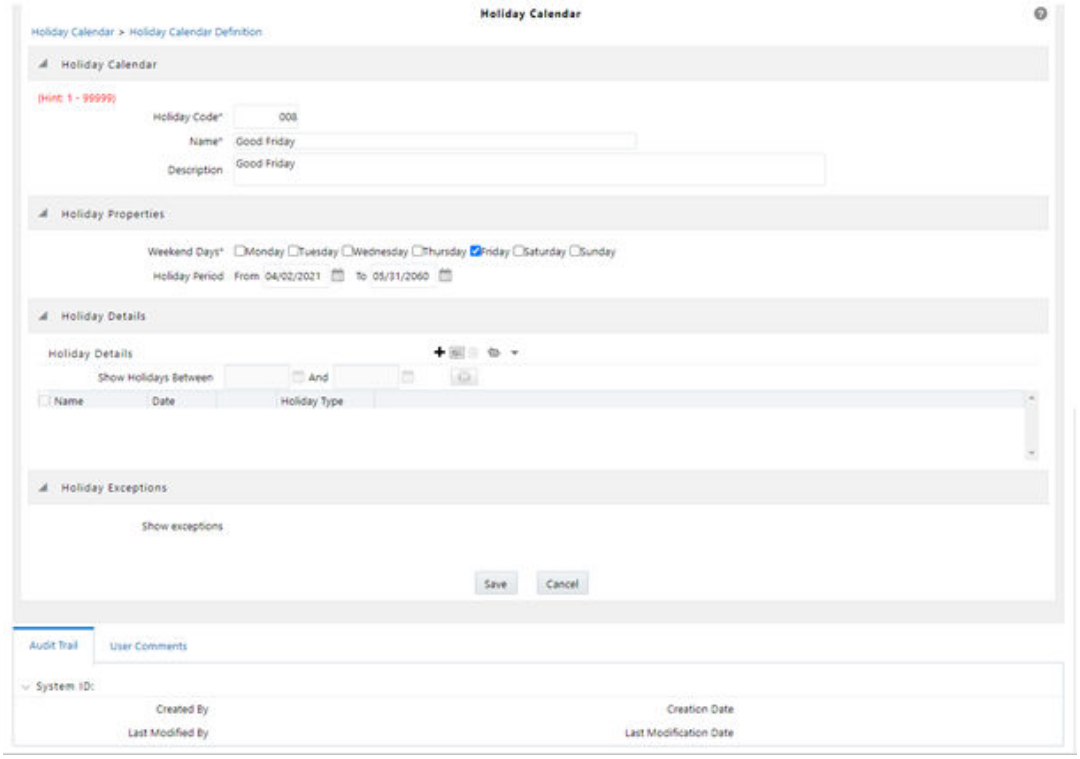
Copyright © 2011, 2026, Oracle and/or its affiliates.

PREVIOUS_WORKINGDAY and NEXT_WORKINGDAY fields designate the valid prior and following working days, respectively.

To create a new Holiday Calendar rule, follow these steps:

1. From the LHS menu, select **Common Object Maintenance**, and then select **Holiday Calendar** to display the **Holiday Calendars** summary page.
2. Click the **Add** icon. The Holiday Calendar Details page is displayed.

Figure 10-2 Holiday Calendar Definition



3. Enter a code value for the new Holiday Calendar.
   The code is a numeric identifier for the Holiday Calendar. The code value must be a unique number between and must be between 1 and 99999.
4. Enter the name and a brief description for the Holiday Calendar.
   The name you assign to the Holiday Calendar must be unique. The name can hold a maximum of 30 characters.
5. In the Holiday Properties grid, select not more than two weekend days. Then choose the Holiday Period. The Holiday Period can be defined for a range of up to 40 years less than the current date and 40 years greater than the current date, totally spanning a maximum of 80 years.
6. In the Holiday Details grid, define the Holiday details for any period within the holiday range defined in the previous step.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> To define a fixed holiday, input the holiday date for the first occurrence in the date range. For example, if your Date Range runs from 01-JAN-2000 to 31-DEC-2050, you should input the fixed holiday, Christmas, as 25-DEC-2000. The Holiday Calendar procedure will populate all subsequent 25-DEC entries in the holiday list table (FSI_HOLIDAY_LIST).

7. Click **Save**. Once the Holiday Calendar definition is saved, its status in the Holiday Calendar Summary page is marked as defined.
8. A Holiday Calendar created can also be deleted. Select one or more check boxes of Holiday Calendar definitions and click **Delete**.

10.3 Excel Import and Export

Excel Import and Export functionality is used for adding or editing Holiday Calendar definitions.

10.4 Executing Holiday Calendar

Execute a Holiday Calendar definition to generate calendar dates listing the various types of holidays for a given holiday period.

Predefined rules are a prerequisite to execute the Holiday Calendar.

To execute the Holiday Calendar, follow these steps:

1. Navigate to the Holiday Calendar Summary page.
2. Search for a rule.
3. Select a Holiday Calendar and Click the Generate Calendar Dates icon to execute the selected Holiday Calendar.

   Holiday list for holiday ID #1 generated successfully message appears (where #1 is the Holiday Calendar code). The holiday list can be confirmed by querying the FSI_HOLIDAY_LIST table.
4. The status of a Holiday Calendar where holiday dates are generated displays as Processed in the status column on the summary page.

   If you do not want to Generate Calendar dates immediately, you can select that particular Holiday Calendar anytime later from the summary page with its status defined, and then click the Generate Calendar Dates icon to execute the selected Holiday Calendar.
5. The generated holiday list is no longer valid if:
   a. There is a change in the definition of the Holiday Calendar.
   b. There is an update or modification to the Holiday Exceptions defined for that holiday calendar.

   In such a case, you will get a message "*This Holiday Calendar has been modified, Please generate the holiday list again*" and the Holiday Calendar state will be changed to Defined until the holiday list is regenerated with a new definition.

10.5 Holiday Exceptions

You can specify exceptions to the holidays.

Copyright © 2011, 2026, Oracle and/or its affiliates.

As a prerequisite, a Holiday Calendar must be properly defined and the status of the Holiday Calendar in the summary page must be processed. Generating the holiday list will populate the holidays (weekends, fixed, and moving) along with the working days. Then the Show Exceptions button is enabled on the detail page. Any changes in the holiday definition will disable the “Show Exceptions” button. The user must generate the holiday list again to define or view the exceptions.

1. Click **Show Exceptions** in the Holiday Exceptions pane. The Holiday Exceptions window opens.
2. The search block in the Exceptions page has the following fields:
   - **From and To**: Denotes the range of years which is a subset out of the holiday list generated, for which exceptions are required to be defined.
   - **Fixed Holidays**: You can filter the list of holidays by the type of Fixed Holidays.
   - **Moving Holidays**: You can filter the list of holidays by the type of Moving Holidays.
   - **Holiday Date**: For a particular known holiday date, exceptions can be defined.
   - **All Exceptions**: This check box when selected lists all the exceptions, if already defined, for the holidays within the From, To Date range.
3. The search result gives the list of all holidays based on the selection of the above search criteria fields.
4. In the Holiday Exceptions pane, there are two types of exceptions that can be defined: Not a holiday and Shift to.
5. Any holiday can be marked as not a holiday, in which case that day is removed from the Holiday List. If you select Not a Holiday from the Exception Type drop-down, then the Shift to date field is disabled.
6. Spring earlier considered as a holiday in the Holiday Calendar can be marked as Not a Holiday in the Holiday Exceptions window. You can write your comments or remarks in the Notes next to the Exception Type drop-down list.
7. Any holiday can be shifted to another day, in which case the earlier declared holiday is removed from the Holiday List, while the shifted day is included as a holiday.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11

OFS Funds Transfer Pricing Overview

Oracle Funds Transfer Pricing (FTP) application is based on the Oracle Financial Services Analytical Applications Infrastructure (OFSAAI). OFSAAI is the central, integrated data source on which Oracle Financial Services Analytical (OFSAA) applications are built. This description of the Oracle Funds Transfer Pricing process assumes that your system administrator has set up the OFSAAI data repository and has populated it with your enterprise-wide business data. For more information, see the OFS Analytical Applications Infrastructure User Guide.

Oracle Funds Transfer Pricing allows you to transfer price instruments, such as mortgages and commercial loans, stored in your Instrument tables, as well as aggregated information, such as cash and other assets, and equity, residing in the Management Ledger table. Consequently, in transfer pricing you need to select the Account tables as the data source for instruments and the Management Ledger Table for aggregated information.

The Oracle Funds Transfer Pricing process comprises the following steps:

> **Tip**
>
> Although the following list of steps is sequential, not all users must follow all of these steps. While some steps might not apply to your product portfolio, others are optional, and you can decide whether to include them to fine-tune your transfer pricing results. All required steps are explicitly marked as mandatory in the list of steps, as well as in the sections where they are described in detail.

- Reconciling the data
- Cleansing the data by Performing Cash Flow Edits
- Setting Application Preferences
- Capturing instrument behavior by:
  - Defining Behavior Patterns
  - Defining Payment Patterns
  - Defining Repricing Patterns
- Activating Currencies and loading exchange rates
- (Mandatory) Deciding on historical rate information and managing it by Creating Interest Rate Codes
- Setting Stochastic Rate Index Rules
- (Mandatory) Defining Transfer Pricing Rules
- Setting Prepayment Model Rules
- Defining Prepayment Rules
- Defining Adjustment Rules
- Defining Alternate Rate Output Mapping Rules
- Creating a Propagation Pattern

Copyright © 2011, 2026, Oracle and/or its affiliates.

- (Mandatory) Defining and Executing the Transfer Pricing Process
- Reviewing Processing Errors
- Accessing Transfer Pricing, Detail Cash Flow Results for Audit Purposes
- Accessing Transfer Pricing, Interest Rate Audit Results
- Analyzing Results
- Reprocessing Erroneous Accounts
- Define Transfer Pricing Rules
  Defining Transfer Pricing rules is a mandatory step in the Oracle Funds Transfer Pricing process. You must define Transfer Pricing rules, to 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.
- Transfer Pricing Methodologies and Rules
  The transfer pricing methodologies supported by Oracle Funds Transfer Pricing can be grouped into the following categories:
- Define Prepayment Rules
  One of the major business risks faced by financial institutions engaged in the business of lending is 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.
- Define Adjustment Rules
  Adjustment Rules allow users to define Transfer Pricing Add-on rates that are assigned incrementally to the base FTP rate to account for a variety of miscellaneous risks such as Liquidity risk or Basis risk or to supplement strategic decision-making with Pricing Incentives, Breakage Charges, or other types of rate adjustments.
- Define Alternate Rate Output Mapping Rules
  Within the Standard Transfer Pricing Process, users can select an appropriate Alternate Rate Output Mapping rule to output transfer pricing results to the seeded or user-defined alternate columns instead of the standard default column.
- Define and Execute a Transfer Pricing Process
  Defining and executing a Transfer Pricing Process is one of the mandatory steps in Oracle Funds Transfer Pricing. The Transfer Pricing Process allows you to:
- Define and Execute a Break Identification Process
  Break Identification processing should be run if automatic break detection is the preferred approach to populating the break events table (FSI_D_BREAK_FUNDING_CHARGES). Users also have the option to populate this table manually during data loading or use a hybrid approach. The Break Funding Charges table is the source table for calculating breakage charges.
- Ledger Migration Process
  This section discusses the process for generating charges or credits, for funds provided or used, and their migration to the Management Ledger table when using the Ledger Level-Charge or Credit migration option in the TP Process.
- Define Behavior Patterns
  A prerequisite for transfer pricing your product portfolio is capturing instrument behavior. Payment and repricing characteristics for most instruments can be accommodated through the data attribute values in the Instrument tables. However, certain instruments may not have contractual cash flows or may have cash flows that are based on unique payment and repricing patterns that are too complex to be accommodated in the standard fields of the Instrument tables. Oracle Funds Transfer Pricing allows you to define custom behavior,

Copyright © 2011, 2026, Oracle and/or its affiliates.

payment, and repricing patterns to accurately model the unique characteristics for such instruments.

- Define Payment Patterns
  A prerequisite for transfer pricing your product portfolio is capturing instrument behavior. Payment and repricing characteristics for most instruments can be accommodated through the data attribute values in the Instrument tables. However, certain instruments may not have contractual cash flows or may have cash flows that are based on unique payment and repricing patterns that are too complex to be accommodated in the standard fields of the Instrument tables. Oracle Funds Transfer Pricing allows you to define custom behavior, payment, and repricing patterns to accurately model the unique characteristics for such instruments.
- Define Repricing Patterns
  User-Defined Repricing Patterns provide a mechanism to capture the repricing structure of instruments whose rates change according to complex schedules which cannot be captured in the standard fields of account tables.
- Perform Cash Flow Edits
  The data in the Instrument tables must be clean, accurate, and complete before it is used to generate cash flows and for further processing. Oracle Funds Transfer Pricing provides seeded Cash Flow Edit Rules to edit (clean and prepare) Instrument table data. You can create multiple Cash Flow Edit Processes depending on the data to be cleansed. You can also view the actual results of Cash Flow Edits by accessing the audit data written into the FSI_PROCESS_ERRORS table.
- Create Interest Rate Codes
  Oracle Funds Transfer Pricing uses historical interest rate information to transfer price your balance sheet. The final transfer rate or add-on rate assigned to the records in your instrument 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 Funds Transfer Pricing implementation.
- Access 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 Audit Block, the detailed cash flow results are written to the FSI_O_PROCESS_CASH_FLOWS table.
- Access Transfer Pricing Interest Rate Audit Results
  Forward Rates and 1 Month Rates audit results can be written to an audit table to facilitate validation of option cost results.
- Analyze Results
  You should always analyze results obtained from the Transfer Pricing Engine. For example, you should review the historical rate information (Interest Rate Codes) to ensure that the new cost of funds reflects the current interest rate reality.
- Review Processing Errors
  There is always the possibility that errors may occur during the execution of a Transfer Pricing Process. A log of such errors is generated during processing and can be accessed by selecting the "View Log" link from both Standard or Stochastic Transfer Pricing Process summary page. Within this log, the report lists the specific transaction for which an error was generated and provides the internally generated identifier of the Transfer Pricing Process that generated it.
- Reprocess Erroneous Accounts
  While reviewing your results, you might discover accounts with invalid results that need to be reprocessed. Transfer Pricing Processes allow you to rerun a subset of information to make corrections to appropriate products when needed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Reconcile the Data
  Reconciliation is the process of comparing the information carried in the Instrument tables to the General Ledger balances (contained in the Management Ledger table).

11.1 Define Transfer Pricing Rules

Defining Transfer Pricing rules is a mandatory step in the Oracle Funds Transfer Pricing process. You must define Transfer Pricing rules, to 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.

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

- **Node Level Assumptions**: Oracle Funds Transfer Pricing uses the Product Dimension that has been selected within Application Preferences to represent a financial institution's product portfolio. Using this dimension, you can organize your product portfolio into a hierarchical structure and define parent-child relationships for different nodes of your product hierarchy. This significantly reduces the amount of work required to define transfer pricing, prepayment, and adjustment rule methodologies.
  You can define transfer pricing, prepayment, and adjustment rule 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. For more information, see Defining Transfer Pricing Methodologies Using Node Level Assumptions.
- **Conditional Assumptions**: The Conditional Assumption 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, Prepayment, or Adjustment rule methods to the fixed-rate loans and another to the floating-rate loans. For more information, see Associating Conditional Assumptions with Assumption Rules.

11.2 Transfer Pricing Methodologies and Rules

The transfer pricing methodologies supported by Oracle Funds Transfer Pricing can be grouped into the following 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.

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 Instrument tables. Consequently, Cash Flow methods apply only if the data source is Account tables. Data stored in the Management Ledger Table reflects only accounting entry positions at a particular point in time and does not have the required financial details to generate cash flows, therefore, preventing you from applying Cash Flow 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Oracle Funds Transfer Pricing supports the following Cash Flow Transfer Pricing methods:

- Cash Flow: Average Life
- Cash Flow: Duration
- Cash Flow: Weighted Term
- Cash Flow: Zero Discount Factors

Non-Cash Flow Transfer Pricing Methods: These methods do not require the calculation of cash flows. While some of the non-cash flow methods are available only with the Account tables data source, some are available with both the Account and Ledger table data sources.

Oracle Funds Transfer Pricing supports the following Non-Cash Flow Transfer Pricing methods:

- Moving Averages
- Straight Term
- Spread from Interest Rate Code
- Spread from Note Rate
- Redemption Curve
- Reference Period's Compounded Rate
- Tractor Method
- Caterpillar
- Weighted Average Perpetual
- Unpriced Account

Oracle Funds Transfer Pricing also allows Mid-period Repricing. This option allows you to take into 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 Non-Cash Flow transfer pricing methods.

Note on Bulk Updates versus Row-by-Row Processing: Any TP method that does not refer to individual account characteristics utilizes a bulk update to assign a single transfer rate to a group of instrument records. Any TP Method that needs to refer to individual account characteristics to process will execute on a row-by-row basis. In general, Bulk updates are faster than row-by-row processing.

The following TP methods, when not defined through a conditional assumption and not utilizing Mid-Period Repricing, use Bulk Updates:

- Redemption Curve (Assignment Date = As-of-Date only)
- Moving Average
- Spread from Note Rate
- Spread from IRC (Assignment Date = As-of-Date only)

All other TP Methods like Tractor, Caterpillar, and Weighted Average Perpetual are processed row-by-row. When Conditional Assumptions or Mid Period Repricing are used, processing will always be row-by-row, regardless of the TP Method selection.

- Cash Flow: Average Life
  The Average Life method determines the average life of the instrument by calculating the effective term required to repay half of the principal or nominal amount of the instrument.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The TP rate is equivalent to the rate on the associated interest rate curve corresponding to the calculated term.

- Cash Flow: Duration
  The Duration method uses the Macaulay Duration Formula:
- Cash Flow: Weighted Term
  The Weighted Term method builds on the theoretical concepts of duration. You can use the Cash flow Duration TP method approach to the Cash Flow Weighted Term method. Based on that, the following Cash Flow Discounting Methods are used:
- Cash Flow: Zero Discount Factors
  The Zero Discount Factors (ZDF) 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-coupons).
- 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 of a particular product.
- Straight Term
  When you select the Straight Term method and Standard Term approach, the system derives the transfer rate using the last repricing date and the next repricing date for adjustable-rate instruments, and the origination date and the maturity date for fixed-rate instruments.
- 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:
- 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.
- 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 in the application with this method:
- Reference Period's Compounded Rate
  This Transfer Pricing method is to support IBOR transition. Preferably IRC should be Risk-free rate compounded index. As per the selected period, system picks the compounded index for the reference period start date and does the calculation till the reference period end date as follows:
- Tractor Method
  The Tractor Method extends the concept of “Strip Funding” to instruments that do not have contractual cash flows.
- Caterpillar
  The Caterpillar method extends the concept of “strip funding” to instruments that do not have contractual cash flows.
- Weighted Average Perpetual
  This method calculates the simple weighted average of the applicable balance based on maturity bands defined through a Behavior Pattern. In the following example, the end of

Copyright © 2011, 2026, Oracle and/or its affiliates.

period balance as of Jan 31 is split into Core and Volatile strips, and the relevant rates are applied to arrive at the weighted average Transfer Rate.

- Unpriced Account
  Under the unpriced account method, the transfer rate for the account is defined as the weighted average of the Product dimension members. 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.
- Risk Free Reference Rates
  Risk Free Reference rates (RFR) could be used for following two methodologies, based on what is selected in UI:
- 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 non-cash flow transfer pricing methods:
- Defining Transfer Pricing Methodologies Using Node Level Assumptions
  In Oracle Funds Transfer Pricing, your product portfolio is represented using the Product Dimension specified in your FTP Application Preferences. Node Level Assumptions allow you to define transfer pricing, prepayment, and adjustment assumptions at any level of the Product dimension Hierarchy. The Product dimension supports a hierarchical representation of your chart of accounts, so you can take advantage of the parent-child relationships defined for the various nodes of your product hierarchies while defining transfer pricing, prepayment, and adjustment assumptions. Child nodes for which no assumptions are specified automatically inherit the methodology of their closest parent node. Conversely, explicit definitions made at a child level will take precedence over any higher-level parent node assumption.
- Associating Conditional Assumptions with Assumption Rules
  Oracle Funds Transfer Pricing extends the setup and maintenance of assumptions by allowing users to integrate conditional logic (optional) into the setup of transfer pricing, prepayment, and TP adjustment methods. The Caterpillar method under Transfer Pricing Rules will not be available for selection under Conditional Assumptions.

11.2.1 Cash Flow: Average Life

The Average Life method determines the average life of the instrument by calculating the effective term required to repay half of the principal or nominal amount of the instrument. The TP rate is equivalent to the rate on the associated interest rate curve corresponding to the calculated term.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-1 Cash Flow: Average Life Formula

$$\text{Average Life} = \sum_{i=1}^{n} \frac{P_i}{P} t_i$$

Where:

P is the principal

P_i is the principal repayment in coupon i, hence

$\frac{P_i}{P}$ is the fraction of the principal repaid in coupon i, and

t_i is the time from the start of coupon i

Oracle Funds Transfer Pricing derives the Average Life based on the cash flows of an instrument as determined by the characteristics specified in the Instrument Table and using your specified prepayment rate, if applicable. The average life formula calculates a single term, that is, a point on the yield curve used to transfer the price of the instrument being analyzed. The Average Life calculation does not differentiate between fixed-rate and adjustable-rate instruments. It applies the same calculation logic to both. It computes the Average Life of the loan (to maturity).

> **Note**
>
> The Average Life TP Method provides the option to Output the result of the calculation to the instrument record (TP_AVERAGE_LIFE). This can be a useful option if you would like to refer to the Average Life as a reference term within an Adjustment Rule.

Users also have the choice to populate the TP_AVERAGE_LIFE column directly with a value computed outside of OFSAA FTP. If this value is populated, the FTP engine reads the TP_AVERAGE_LIFE and will look up the FTP rate for the given term. In this case, the TP Engine does not generate cash flows and will not re-compute the Average Life. It simply uses the value that has been provided and lookup the appropriate FTP rate from the specified TP Interest Rate Curve.

> **Note**
>
> Repricing patterns are now supported for the Cashflow Average Life method. For index repricing patterns, rates will be referred from the specified IRC based on the account origination date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.2.2 Cash Flow: Duration

The Duration method uses the Macaulay Duration Formula:

Figure 11-2 Cash Flow: 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.
- CFn: Cash flow (such as Regular Principal, Prepayments, and Interest) in period n.
- r: Periodic Rate (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 Funds Transfer Pricing derives the Macaulay duration based on the cash flows of an instrument as determined by the characteristics specified in the Instrument 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.

- Within the Duration Calculation, the discount rate or current rate, r, is defined in one of three ways, based on how the methodology is set up by the user:
  - The current rate is defined as the Current Net Rate if the processing option, "Model with Gross Rates" is not selected and the Current Gross Rate if the option is selected. The current rate is used as a constant discount rate for each cash flow.
  - The user may directly input while defining the TP rule, a constant rate to use for discounts. If specified, this rate is used as a constant discount rate for each flow.
  - The user can select to discount the cash flows using spot rates from a selected interest rate curve. With this approach, a discount rate is read from the selected interest rate curve corresponding to the term of each cash flow.

> **Note**
>
> The Duration TP Method provides the option to Output the result of the calculation to the instrument record (TP_DURATION). This can be a useful option if you would like to refer to the duration as a reference term within an Adjustment Rule.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Users also have the choice to populate the TP_DURATION column directly with a value computed outside of OFSAA FTP. If this value is populated, the FTP engine reads the TP_DURATION and will look up the FTP rate for the given term. In this case, the TP Engine does not generate cash flows and will not re-compute the DURATION. It simply uses the value that has been provided and look up the appropriate FTP rate from the specified TP Interest Rate Curve.

11.2.3 Cash Flow: Weighted Term

The Weighted Term method builds on the theoretical concepts of duration. You can use the Cash flow Duration TP method approach to the Cash Flow Weighted Term method. Based on that, the following Cash Flow Discounting Methods are used:

- Multiple Rate
- Single Rate

For more information, see the Transfer Pricing Rules section.

As shown earlier, duration calculates a weighted-average term by weighting each 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 Weighted Term method is to calculate a weighted average transfer rate, it weights the transfer rate in each period, yn, 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. Long-term cash flows have a relatively larger impact on the average transfer rate. The Weighted Term method, with Discounted Cash Flow option selected, can be summarized by the following formula:

Figure 11-3 Cash Flow: Weighted Term 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
- CFn: Cash Flow (such as Regular Principal, Prepayments, and Interest) in period n
- r: Periodic Rate (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
- yn: Transfer rate in period n

Within the Cash Flow Weighted Term method definition screen, users can select the Cash Flow type as either Principal + Interest (the default selection) or Principal Only. This selection impacts the CFn in the above formula.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Additionally, users can choose whether or not to discount the cash flows as described above. If the "Cash Flow" option is selected rather than "Discounted Cash Flow", the following simplified formula is applied:

Figure 11-4 Cash Flow: Weighted Term Formula without Discounted Cash Flow

$$\text{Weighted Average} - y - \left(\frac{\Sigma_{n=1}^{N} \left[y_n \times CF \times t_n \right]}{\Sigma_{n-1}^{N} \left[CF \times t_n \right]} \right)$$

For this method, the following options are also provided:

- Cash Flow Weighted Rate
- Cash Flow and Terms Weighted Rate (by default, this will remain selected for all existing definitions)

If Transfer Rate needs to be weighed only by Cash Flow rather than both Cash Flow and the Cash Flow Weighted Rate can be selected, and the system will not consider terms (tn) for calculations.

If Cash Flow and Terms Weighted Rate is selected, then both terms, as well as Cash Flow, will be used for Weighing Transfer Rate as per calculations shown above.

The discount rate or current rate, r, is defined in one of three ways, based on how the methodology is set up by the user:

- The current rate is defined as the current net rate from the instrument record unless the processing option, "Model with Gross Rates" is selected, in which case, the current gross rate is used. The current rate is used as a constant discount rate for each cash flow.
- The user may directly input while defining the TP rule, a single constant rate to use for discounts. If specified, this rate is used as a constant discount rate for each cash flow.
- The user can select to discount the cash flows using spot rates from a selected interest rate curve. With this approach, a discount rate is read from the selected interest rate curve corresponding to the term of each cash flow.

> **Note**
>
> When validating the Cash Flow Weighted Term transfer rate, FE 492 (Discount Factor) from detail cash flow output is useful. FE 490 (Discount Rate) however, may be incorrect in the detailed cash flow output if the Current Net Rate is specified as the discount rate. This condition does not affect the accuracy of the calculated discount factor, only the audit table rate output for FE 490. If multiple rate discounting (based on IRC) or a single custom rate is specified, then FE 490 will be correct. For more information on FEs, see the OFS Cash Flow Engine Reference Guide.

11.2.4 Cash Flow: Zero Discount Factors

The Zero Discount Factors (ZDF) 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-coupons).

Copyright © 2011, 2026, Oracle and/or its affiliates.

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).

ZDF 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, ZDF 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 ZDF 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.

ZDF 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 the 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.

Customer can set IRCACCRUAL_DSNTFACTOR_FLAG as ‘Y’, in SETUP_MASTER table; in case you want to use IRC accrual for discount factor calculations. By default, system always converts IRC to A/A before it is used for discount factor calculations.

Execute the following SQL statement to set IRCACCRUAL_DSNFACTOR_FLAG to Y:

```
Insert into SETUP_MASTER
(V_COMPONENT_CODE,V_COMPONENT_DESC,V_COMPONENT_VALUE) values
('IRCACCRUAL_DSNTFACTOR_FLAG','IRCACCRUAL_DSNTFACTOR_FLAG','Y');
```

Figure 11-5 Zero Discount Factors = y =

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

In this formula:

- B0: Beginning balance at the time, 0
- Bn-1: Ending balance in the previous period
- Bn: Ending balance in the current period
- DTPn: Discount factor in period n based on the TP yield curve

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 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)

> **Note**
>
> While calculating the number of payments per year, the system can consider the accrual factor defined at the account level when the **Consider Account Accrual Factor** checkbox is enabled.

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

Table 11-1 Deriving Zero-Coupon Discount Factors: An Example

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)/100 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

Indeed, the discount factor is calculated using the following formula:

```
1/ ((1 + Discount rate/100)^(Annualized Discount Term) )
```

For more details, you can refer to OFS Cash Flow Engine Reference Guide.

> **Note**
>
> For the ZDF method, the discount factor used for discounting cash flows is output as FE 490, after multiplied by 100. For more information on FEs, see the OFS Cash Flow Engine Reference Guide.

11.2.5 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 of a particular product.

The following options become available on the user interface (UI) along with Arithmetic and Geometric mean, with this method:

- **Interest Rate Code**: Select the Interest Rate Code to be used as the yield curve to generate transfer rates.
- **Assignment Date**: The Assignment Date allows you to choose the date from which the Moving Average will be calculated. Choices available are the As of Date, Last Repricing

Copyright © 2011, 2026, Oracle and/or its affiliates.

Date, Origination Date, Commitment Start Date, TP Effective Date, or Adj Effective Date. If the selected date is null or 01-Jan-1900, then As of Date will be used as fallback logic.

- **Yield Curve Term**: The Yield Curve Term defines the point on the Interest Rate Code that is used.
- **Historical Range**: 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.

Table 11-2 Yield and Historical Terms: An Example

Moving Average	Yield Curve Term	Historical Range
Six-month moving average of the 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 As-of-Date minus the Historical Term plus one, because the Historical Term includes the As-of-Date.

Table 11-3 Assignment Date and Transfer Rate Calculation

Icons	Description
As Of Date	If the As-of-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. As of Date is available only of the selected source is Ledger table.
Last Repricing Date	If the Last Repricing 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.
Origination Date	If the Origination 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.
Commitment Start Date	If the Commitment Start 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.
TP Effective Date	If the TP 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-3 (Cont.) Assignment Date and Transfer Rate Calculation

Icons	Description
Adjustment Effective Date	If the Adjustment 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**
>
> If any of the above date values are 01-Jan-1900 or blank or null, then the OFS FTP engine considers As-of-Date for Transfer Rate calculation.

After you select the date, OFS FTP computes the Historical Rate using the selected date. OFS FTP takes the values of the yield curve points that fall within that range and does a straight average.

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

11.2.6 Straight Term

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

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

    > **Note**
    >
    > For loans in the Tease period, the Next Reprice Date should reflect the end of the Tease Period and the reprice frequency should reflect the expected reprice frequency after the tease period ends.

- Remaining Term Calculation Mode:

Copyright © 2011, 2026, Oracle and/or its affiliates.

  - For Fixed Rate Products, use As-of-Date and Maturity - As-of-Date.
  - For Adjustable Rate Products, use As-of-Date and Next Repricing Date - As-of-Date. In addition to the standard logic used for determining the appropriate “Term”, users also have the option to select either Original Term or Repricing Frequency and also have the option to modify these terms using simple mathematical operators. These options can be useful in cases where the straight term method should be applied to the same record under different circumstances. For example, for calculating the base rate on an adjustable-rate instrument, the standard approach should be used. For the same instrument, users may further want to use the entire original term for applying a liquidity premium or other add-on rate. To support the second case, we give the option to directly specify the term to be used, and we further provide the option to modify the term using simple operators, such as +, -, *, /.

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

- **Term**: Select from Standard, Original Term, or Reprice Frequency. Standard is the default selection and the resulting Term will follow the above logic. The Original Term and Reprice Frequency options allow users to override the standard logic and specify which term to use.
- **Adjustment Operator**: When either Original Term or Reprice Frequency is selected as the Term, the Adjustment Operator becomes active. The term adjustment is optional and gives users the ability to modify the term.
- **Adjustment Amount**: This input works together with the adjustment operator to indicate how the term should be modified.
- **Interest Rate Code**: Select the Interest Rate Code to be used for transfer pricing the account.
- **Mid-Period Repricing Option**: Select the checkbox beside this option to invoke the Mid-Period Repricing option.
- **Holiday Calendar**: Select whether a holiday calendar is applicable for calculating the charges/credits or for calculating Economic Value.
- **Rolling Convention**: Select the appropriate business day rolling convention if a Holiday Calendar is selected.
- **Interest Calculation Logic**: Select the appropriate option to indicate how the interest payment should be adjusted when a holiday date is encountered.

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

For more information, see the Transfer Pricing Rules section.

11.2.7 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

Copyright © 2011, 2026, Oracle and/or its affiliates.

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 before 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 As-of-Date, Last Repricing Date, Origination Date, Adjustment Effective Date, or TP Effective 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: Ledger Table or Account Tables.

11.2.8 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 appropriately 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 should enter a negative rate spread.

Table 11-4 Example of Rate Spread

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

The following option becomes available in 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.2.9 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 in 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 As-of-Date, Last Repricing Date, Origination Date, Adjustment Effective Date, or TP Effective Date.
- **Percentages or Term Points**: See Redemption Curve.
- **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: Ledger Table or Account Tables.

11.2.10 Reference Period's Compounded Rate

This Transfer Pricing method is to support IBOR transition. Preferably IRC should be Risk-free rate compounded index. As per the selected period, system picks the compounded index for the reference period start date and does the calculation till the reference period end date as follows:

For this method configuration, along with the IRC selection, a term point selection is also given, which will be by default set at a minimum available term point.

Reference Period Start Date:

- Last Payment Date (Default)
- Origination Date
- Last Reprice Date
- TP Effective Date
- Adjustment Effective Date

Reference Period End Date will always be As of Date.

Table 11-5 Reference Period's Compounded Rate

Date	SONIA Compound Index
02 Jun 21	101.34331802
01 Jun 21	101.34317974
28 May 21	101.34260667
27 May 21	101.34246757
26 May 21	101.34232875
25 May 21	101.34219103

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-5 (Cont.) Reference Period's Compounded Rate

24 May 21	101.34205304
21 May 21	101.34164073
20 May 21	101.34150385
19 May 21	101.34136780
18 May 21	101.34123092
17 May 21	101.34109682
14 May 21	101.34068285
13 May 21	101.34054569
11 May 21	101.34027360
10 May 21	101.34013756
07 May 21	101.33972276
06 May 21	101.33958643
05 May 21	101.33945122
04 May 21	101.33931351
30 Apr 21	101.33874824
29 Apr 21	101.33860969

Where:

- x = 30th April 2021
- y = 28th May 2021
- d = 28

Figure 11-6 Compounded SONIA rate

$$\text{Compounded SONIA rate between x and y} = \left(\frac{\text{SONIA Compounded Index}_y}{\text{SONIA Compounded Index}_x} - 1 \right) X \frac{365}{d}$$

$$\text{Compounded SONIA rate between 30th April 2021 and 28th May 2021} = \left(\frac{101.34260667}{101.33874824} - 1 \right) X \frac{365}{28}$$

Compounded SONIA rate between 30th April 2021 and 28th May2021 = 0049633% (6 dp)

11.2.11 Tractor Method

The Tractor Method extends the concept of “Strip Funding” to instruments that do not have contractual cash flows.

These products are known as perpetual or non-maturity products and hence do not generate contractual cash flows. The process of determining transfer rates requires adopting the strip funding approach by splitting these products into Core and Volatile portions based on statistically established Behavioral Profiles. With this method, the volatile portion is considered to have an overnight maturity and the core portion is assigned a longer maturity (through a Behavior Pattern) which is comprised of a series of balance strips corresponding to the

Copyright © 2011, 2026, Oracle and/or its affiliates.

maximum tenor of the portfolio (on the origination of the portfolio, original strips will have shorter maturities). Each strip is assigned a transfer rate corresponding to its origination date and corresponding term. The historical, active strips are retained for the portfolio and the portfolio transfer rate is determined based on the weighted average transfer rate of the strips comprising the portfolio. The weighted average rate from the strip portfolio is written back to all instrument records that are mapped to this portfolio through the Transfer Pricing rule.

The characteristics of the replicating portfolio used by the Tractor Method are defined through a Non-Maturity Behavior Pattern with replicating portfolio options. The following example illustrates the behavior of a small 3-day portfolio rolling across 4 days.

Table 11-6 Replicating Portfolio

31-Jan-11						
Strip	**Start**	**Maturity**	**Balance**	**TP Rate**	**Date Rolled**	**Comment**
0A	31-Jan-11	1-Feb-11	100	5.00%	1-Feb-11	
0B	31-Jan-11	2-Feb-11	100	5.10%	3-Feb-11	
0C	31-Jan-11	3-Feb-11	100	5.20%	3-Feb-11	Matures and rolls to 3A
1A	1-Feb-11	4-Feb-11	100	5.30%		
2A	2-Feb-11	7-Feb-11	100	5.40%		
3A	3-Feb-11	8-Feb-11	100	5.50%		Rollover strip, TP's by the engine
Day 1						
1-Feb-11	Daily Rollover					
Strip	Start	Maturity	Balance	Rate	Date Rolled	Comment
0A	31-Jan-11	1-Feb-11	100	5.00%	1-Feb-11	Matures and rolls to 1A
0B	31-Jan-11	2-Feb-11	100	5.10%		
0C	31-Jan-11	3-Feb-11	100	5.20%		
1A	1-Feb-11	4-Feb-11	100	5.30%		Rollover strip, TP's by the engine
	Day 1 Maturity Profile					
	Total	Day 1	Day 2	Day 3		
		2-Feb-11	3-Feb-11	4-Feb-11		Tractor TP Rate
	300	100	100	100		5.20%
Day 2						
2-Feb-11	Daily Rollover					
0A	31-Jan-11	1-Feb-11	100	5.00%	1-Feb-11	
0B	31-Jan-11	2-Feb-11	100	5.10%	2-Feb-11	Matures and rolls to 2A
0C	31-Jan-11	3-Feb-11	100	5.20%		
1A	1-Feb-11	4-Feb-11	100	5.30%		
2A	2-Feb-11	7-Feb-11	100	5.40%		Rollover strip, TP'd by the engine

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-6 (Cont.) Replicating Portfolio

31-Jan-11						
Strip	**Start**	**Maturity**	**Balance**	**TP Rate**	**Date Rolled**	**Comment**
	Day 2 Maturity Profile					
	Total	Day 1	Day 2	Day 3		
		3-Feb-11	4-Feb-11	7-Feb-11		Tractor TP Rate
	300	100	100	100		5.30%
Day 3						
3-Feb-11	Daily rollover, and new business (reduction of 530)					
0A	31-Jan-11	1-Feb-11	100	5.00%	1-Feb-11	
0B	31-Jan-11	2-Feb-11	100	5.10%	3-Feb-11	
0C	31-Jan-11	3-Feb-11	100	5.20%	3-Feb-11	Matures and rolls to 3A
1A	1-Feb-11	4-Feb-11	100	5.30%		
2A	2-Feb-11	7-Feb-11	100	5.40%		
3A	3-Feb-11	8-Feb-11	100	5.50%		Rollover strip, TP'd by the engine
3B	3-Feb-11	4-Feb-11	10	5.50%		New Business strip, TP'd by the engine
3C	3-Feb-11	7-Feb-11	10	5.50%		New Business strip, TP'd by the engine
3D	3-Feb-11	8-Feb-11	10	5.50%		New Business strip, TP'd by the engine
	Day 3 Maturity Profile					
	Total	Day 1	Day 2	Day 3		
		4-Feb-11	7-Feb-11	8-Feb-11		Tractor TP Rate
	270	90	90	90		5.39%
Day 4						
4-Feb-11	Daily rollover					
0A	31-Jan-11	1-Feb-11	100	5.00%	1-Feb-11	
0B	31-Jan-11	2-Feb-11	100	5.10%	2-Feb-11	
0C	31-Jan-11	3-Feb-11	100	5.20%	3.-Feb-11	
1A	1-Feb-11	4-Feb-11	100	5.30%	4-Feb-11	Matures and rolls to 4A

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-6 (Cont.) Replicating Portfolio

31-Jan-11						
Strip	**Start**	**Maturity**	**Balance**	**TP Rate**	**Date Rolled**	**Comment**
2A	2-Feb-11	7-Feb-11	100	5.40%		
3A	3-Feb-11	8-Feb-11	100	5.50%		
3B	3-Feb-11	4-Feb-11	-10	5.50%	4-Feb-11	Matures and rolls to 4A
3C	3-Feb-11	7-Feb-11	-10	5.50%		
3D	3-Feb-11	8-Feb-11	-10	5.50%		
4A	4-Feb-11	9-Feb-11	90	5.60%		Rollover strip, TP'd by the engine
	Day 4 Maturity Profile					
	Total	Day 1	Day 2	Day 3		
		7-Feb-11	8-Feb-11	9-Feb-11		Tractor TP Rate
	270	90	90	90		5.50%

This example assumes a business day calendar is enabled with the following business days defined:

- Business Day
- Calendar
- 31-Jan-11
- 1-Feb-11
- 2-Feb-11
- 3-Feb-11
- 4-Feb-11
- 7-Feb-11
- 8-Feb-11
- 9-Feb-11
- 10-Feb-11
- 11-Feb-11

11.2.12 Caterpillar

The Caterpillar method extends the concept of "strip funding" to instruments that do not have contractual cash flows.

These products are known as Perpetual or Non-Maturity Products and therefore do not generate contractual cash flows. The process of determining Transfer Rates requires adopting the 'Strip Funding Approach' by splitting these products into 'Core' and 'Volatile' portions based on statistically established Behavioral Profiles. With this approach, the volatile portion may be considered as an overnight funding strip and the core portion can be dealt with by an assumed

Copyright © 2011, 2026, Oracle and/or its affiliates.

maturity structure defined through a Behavior Pattern. For example, statistical analysis may imply that the Savings Account portfolio behaves 20% as volatile and 80% as the core of which the maturity is likely to be 3 months. Therefore, the Funding Strips that would get created are 20% 1 month, and 80% 3 months.

Table 11-7 BEHAVIORAL PATTERN: 20% 1 Month; 80% 3 Months

		Production			IRC			
Period	**EOP Balance**	**1M**	**2M**	**3M**	**1M**	**2M**	**3M**	**TP Rate**
Month 1	4000	800		3200	4.00%	4.25%	4.50%	4.40%
Month 2	7000	760	3200	3040	4.50%	4.75%	5.00%	4.72%
Month 3	8500	3652	3040	1808	5.00%	5.25%	5.50%	4.92%
Month 4	9000	3870	1808	3322	5.50%	5.75%	6.00%	5.52%
Month 5	8000	2382	3322	2296	6.00%	6.25%	6.50%	6.03%

In the example, we assume that once a funding strip is assigned a certain Transfer Rate based on its original term, the rate remains constant until the strip matures. Each strip is funded for the original term based on the yield curve in effect at the start of the strip. In month 4, when the balance is 9000, the strips still outstanding from earlier months are 3040 as a 3-month term strip, created in month 2 at 5% having a remaining term 1-month; (3870-3040) 1-month term strip created in month 4, 1808 3-month term strip created in month 3 with the 2-month remaining term, and 3322 3-month term strip created in month 4. The weighted average rate of these strips comes to 5.52% as the example shows.

In summary: Month 4 Transfer Rate = (3040*5% + (3870-3040)*5.5% + 1808*5.5% + 3322*6%) / 9000 = 5.52%

> **Note**
>
> The Caterpillar Method must not be run more than once for a given date as this may corrupt the historical data. The strip data for this method is stored in the database in the FSI_TP_CATERPILLAR_INTRMDIATE table.

11.2.13 Weighted Average Perpetual

This method calculates the simple weighted average of the applicable balance based on maturity bands defined through a Behavior Pattern. In the following example, the end of period balance as of Jan 31 is split into Core and Volatile strips, and the relevant rates are applied to arrive at the weighted average Transfer Rate.

Table 11-8 BEHAVIORAL PATTERN: 20% 1 Month; 80% 3 Months

Run Date	EOP Balance	1M Strip	3M Strip	1M Rate	3M Rate	TP Rate
Jan 31	100000	20000	80000	3.10%	3.50%	3.42%
Feb 28	200000	40000	160000	3.25%	3.60%	3.53%

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-8 (Cont.) BEHAVIORAL PATTERN: 20% 1 Month; 80% 3 Months

Run Date	EOP Balance	1M Strip	3M Strip	1M Rate	3M Rate	TP Rate
Mar 31	300000	60000	240000	3.20%	3.55%	3.48%

```
Transfer Rate = (20000*3.10% + 80000*3.50%) / 100000 = 3.42%
```

- **Behavior Pattern**: Select the Behavior Pattern that is associated with the Product / Currency combination being defined.
- **Interest Rate Code**: Select the Interest Rate Code, which you want to use as the transfer pricing yield curve.

> **Note**
>
> The Weighted Average Perpetual method applies to both instrument balances and ledger balances.

11.2.14 Unpriced Account

Under the unpriced account method, the transfer rate for the account is defined as the weighted average of the Product dimension members. 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 in the application with this method:

- **Add Dimension Values**: This allows you to select the Product dimension members whose weighted average transfer rate will be assigned to the product being defined.

  > **Note**
  >
  > You should not base an unpriced account on another unpriced account.

- **Across all Organization Units**: This allows you to specify whether the weighted average of transfer rates should be taken across all organizational units. If this option is not selected, the weighted average rate will be calculated for each org unit. To calculate for each individual Org Unit, you must also select the Org Unit dimension under Migration Dimensions in the Transfer Pricing Process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Unpriced Account Method applies only to accounts that use the Ledger Table as their Data Source.
> Users must provide assumptions on the TP Process, "Migration" screen, choosing applicable dimensions, while using Unpriced Account Method irrespective of Ledger Migration being selected or not.
>
> If any dimension is selected as Migration Dimension like GL Account ID, Weighted Average Rate would consider that too and WAR/Charge credit will only be calculated for combinations which are present in both Source and Target Product IDs'.
>
> If Source and Target Product IDs' have different values for Migration Dimensions like GL Account ID, then do not select that dimension (GL Account ID) as Migration Dimension."
>
> For more information on Migration screen parameters, see the Transfer Pricing Process chapter.

11.2.15 Risk Free Reference Rates

Risk Free Reference rates (RFR) could be used for following two methodologies, based on what is selected in UI:

- Cumulative Compounded Rate
- Non-Cumulative Compounded Rate

Step 1: Unannualized/ Effective RFR:

$K_i = \frac{r_i \times n_i}{N}$

N = It will be as per accrual Basis on the record. E.g. if 1 - Then 360 days in an year. ri is the RFR rate for that day and ni - is number of days in interest period.

Step 2: Compounding Factor:

$L_i = \prod_{i=1}^{d_b}\left(1+\frac{r_i \times n_i}{N}\right)$

= (1+ K_i) * (L_i-1BD) - Daily Compounding Factor, if user is directly loading compounding factor, then Step 1 and 2 are skipped.

Step 3: Annualized Cumulative Compounded RFRi (ACRi):

$ACR_i = \left[\prod_{i=1}^{d_b}\left(1+\frac{r_i \times n_i}{N}\right)-1\right]\times\frac{N}{tn_i}$ = N is days in an Year on Accrual Basis CD. tn_i = Cumulative days till nth daily interest period. Here, system performs rounding, based on selected decimal Points in UI. In UI, decimal points are given in % format.

Step 4: Unannualized Cumulative Compounded RFRi:

Copyright © 2011, 2026, Oracle and/or its affiliates.

$UCR_i = ACR_i \times \frac{tn_i}{N}$

Step 5: Non-Cumulative Compounded RFRi:

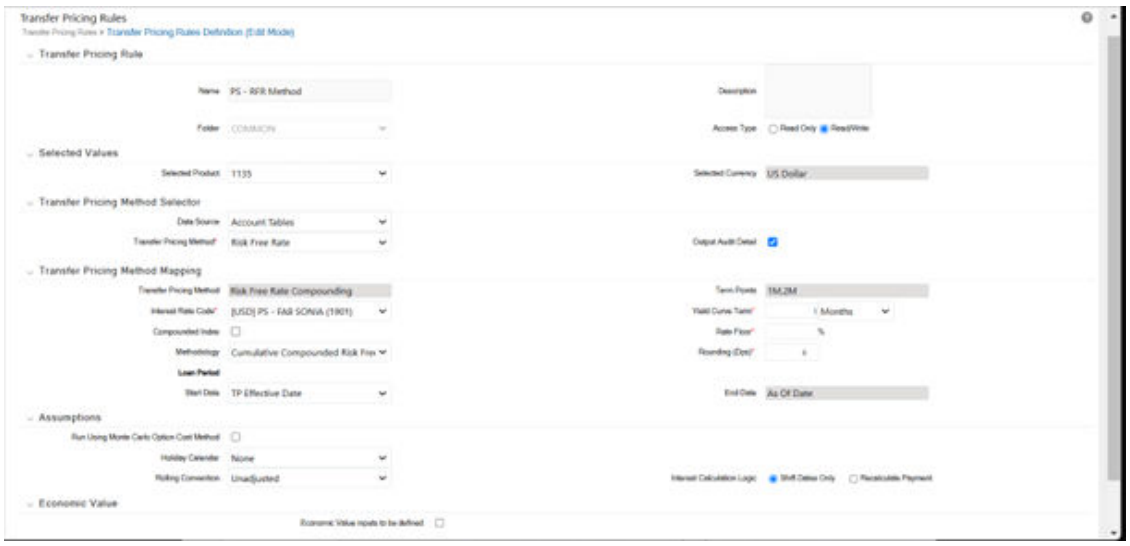
$(UCR_i - UCR_{i-1BD}) \times \frac{N}{n_i}$

n_i is the Number of days in each respective interest period.

N - Number of Days in a year as per the account's accrual Basis CD.

Lag/Lookback will be from - RATE_SET_LAG and RATE_SET_LAG_MULT.

Figure 11-7 Transfer Pricing Rules



Period Start Date: Any of following:

- Last Payment Date (Default)
- Origination Date
- Last Reprice Date
- TP Effective Date
- Add-On Effective Date

Period End Date: As of Date (Maturity Date if it is before As of Date)

If the user selects Cumulative Compounded RFR, then Step 3 will be the output in percentage format and stops.

If the user selects Non-Cumulative Compounded RFR, Step 5 will be the output in percentage format.

Rate Floor (in Percentage): It accepts only integers, between -99.99% to + 99.99%. If RFR is below this rate for any day, the RFR will be used as floor for that date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

To form the Loan/Interest Period, select a holiday from a record. If the holiday is not mentioned in the record, then the system uses whichever day defined in the UI. If nothing is defined, the system considers each day as working day.

Interest Rate Code: A list of all the available IRC for selected CCY or for all CCY curves is displayed and all the yield curve term points for the selected curve are displayed in display only mode.

Yield Curve Term: Enter any integer and multiplier. If user selects term point that is not available on the curve, then system will do the regular interpolation/extrapolation.

If the system searches for particular effective date in IRC and does not find the same, it checks for most recent past available effective date for same curve. If a most recent past effective date for the curve is not found, then system uses the most recent future available effective date for same curve.

Rounding Dps: (Decimal Point for (Annualized Cumulative Compounded Rate) = Integer Inputs. Enter the value in percentage (%).

11.2.16 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 non-cash flow transfer pricing methods:

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

The rationale behind mid-period repricing is as follows. If you do not select the Mid-Period Repricing option, Oracle Funds 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, which reprices on the 15th of the month, and that transfer rates have moved up 200 basis points since the last reprice. In this 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. 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 Instrument 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.
- Typical Calculations
  The following diagram depicts a typical Mid-Period Repricing situation where the instrument reprices during the current processing month.
- Exceptions to Typical Calculations
  There are two exceptions to typical mid-period repricing computations:

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.2.16.1 Mid-Period Repricing Computations

The Mid-Period Repricing option uses two columns in the Instrument 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.

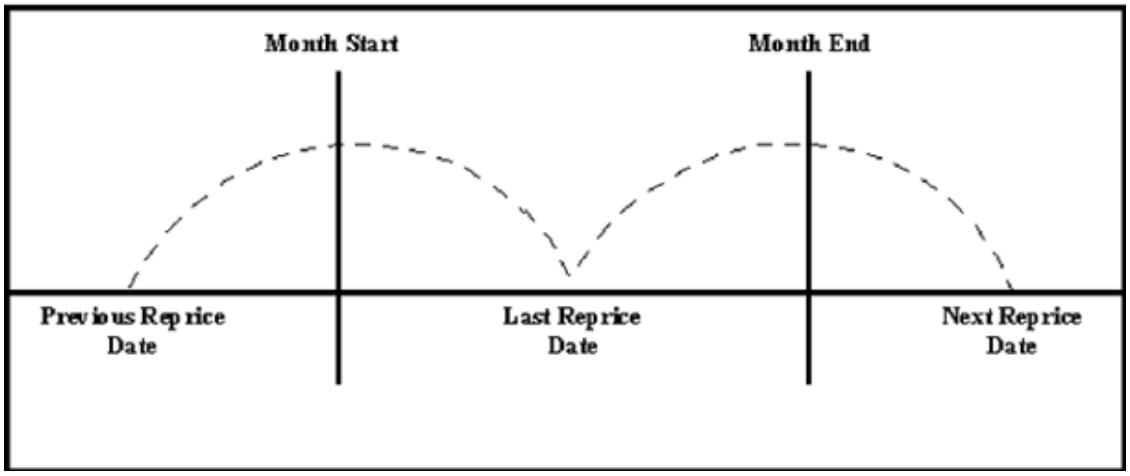
The Mid-Period Repricing Computation process comprises the following steps:

1. Computation of transfer rate for the current repricing period.
2. If the computed Last Repricing Date is less than (>) the beginning of the processing month, then roll back to the 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.

11.2.16.2 Typical Calculations

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

Figure 11-8 Mid-Period Repricing



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:

- Computation of transfer rate for the current repricing period.
  - Transfer Pricing Term: Next Reprice Date - Last Reprice Date

Copyright © 2011, 2026, Oracle and/or its affiliates.

  - Transfer Pricing Date: Last Reprice Date
  - Number of Days at that Rate: End of Month + 1- Last Reprice Date

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, the 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.

Table 11-9 Logic of 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

- 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)

- Computation of the 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 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, the Number of Days equals Minimum (End of Month + 1, Computed Next Reprice Date) - Maximum (Beginning of Month, Computed Last Reprice Date).
  - 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.
  - Computation of the final transfer rate by weighting the results (from current and previous repricing periods) by average balances and days.

The calculation makes the following assumptions:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 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.
- Application of the final transfer rate to the instrument record.

11.2.16.3 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 the term equal to the TEASER_END_DATE - ORIGINATION_DATE, date equals the ORIGINATION_DATE.
  - When rolling backward 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 Funds 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 Funds Transfer Pricing were to shorten the number of days (as in the weighted average calculation), it would double-count the impact.

The following table displays a situation where the Origination Date occurs during the processing month:

Table 11-10 Origination Date Exception: An Example

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.

Considering the origination date exception, the Mid-Period Repricing calculation is done as follows:

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

Copyright © 2011, 2026, Oracle and/or its affiliates.

If period 1 was not taken into account, the result will be, (6% * $50 * 10 days) + (8% * $100 * 10 days) / ($50 * 10 days + $100 * 10 days) = 7.33%, which is incorrect.

11.2.17 Defining Transfer Pricing Methodologies Using Node Level Assumptions

In Oracle Funds Transfer Pricing, your product portfolio is represented using the Product Dimension specified in your FTP Application Preferences. Node Level Assumptions allow you to define transfer pricing, prepayment, and adjustment assumptions at any level of the Product dimension Hierarchy. The Product dimension supports a hierarchical representation of your chart of accounts, so you can take advantage of the parent-child relationships defined for the various nodes of your product hierarchies while defining transfer pricing, prepayment, and adjustment assumptions. Child nodes for which no assumptions are specified automatically inherit the methodology of their closest parent node. Conversely, explicit definitions made at a child level will take precedence over any higher-level parent node assumption.

Node level assumptions simplify the process of applying rules in the user interface and significantly reduce the effort required to maintain business rules over time as new products are added to the product mix. It is also not required for all rules to assign assumptions to the same nodes. Users may assign assumptions at different levels throughout the hierarchy.

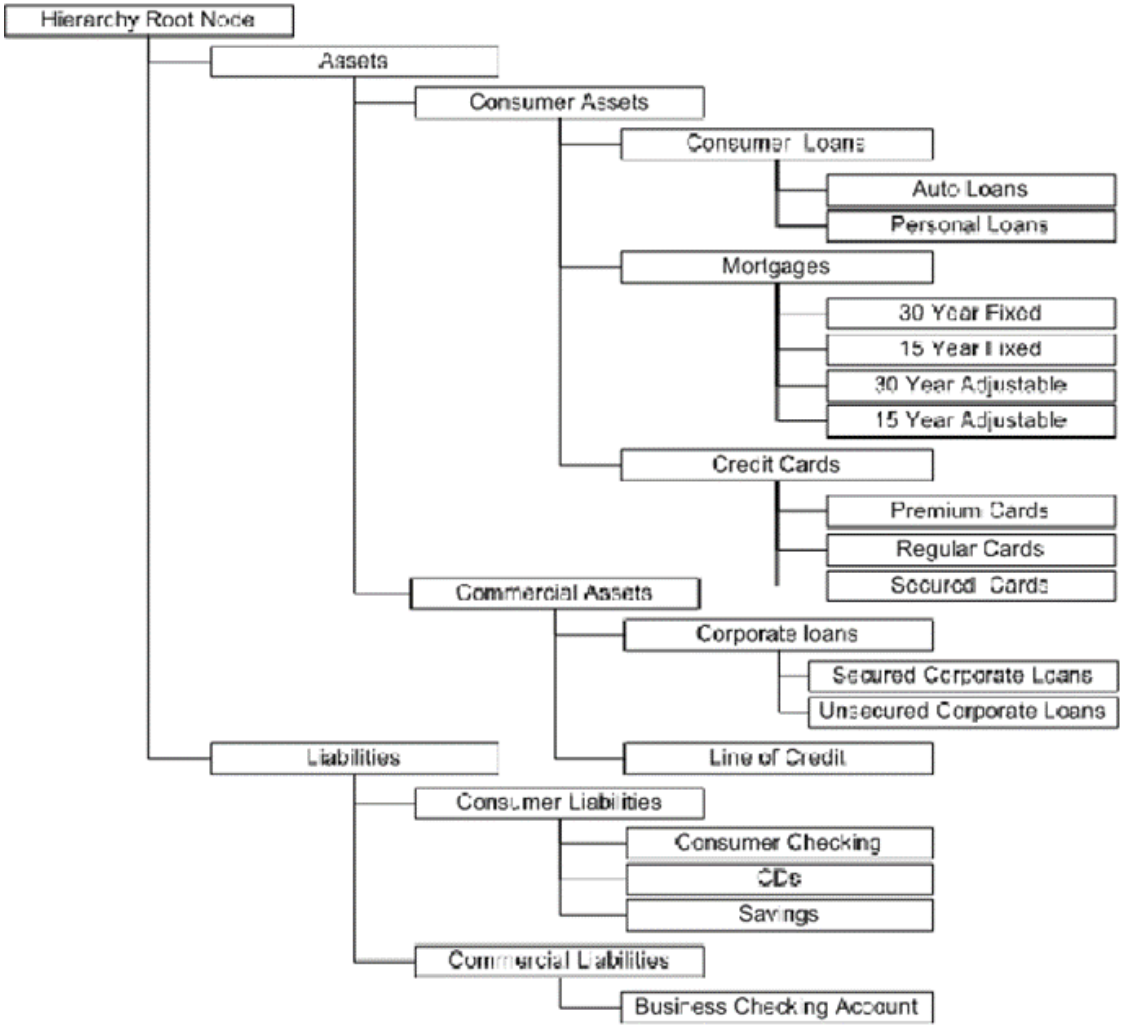
> **Note**
>
> While creating a new rule, if you perform any activities (such as conditional assumption creation, defining products, search, copy across, and so on) in the Assumption window and click the Cancel button, the Rule will be saved with basic Rule definition and displayed in Rule summary page.

The Behavior of Node Level Assumptions:

The following graphic displays a sample product hierarchy:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-9 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 Discount Factors cash flow based 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 Discount Factors 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 previous example. Since the Spread from IRC is specified at the leaf level, the system does not need to search any further to

Copyright © 2011, 2026, Oracle and/or its affiliates.

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.

> **Note**
>
> Not specifying assumptions for a node is not the same as selecting the "Do Not Calculate" method. Child nodes for which no assumptions are specified automatically inherit the methodology of their closest parent node. So if neither a child node nor its immediate parent has a method assigned, the application searches up the nodes in the hierarchy until it finds a parent node with a method assigned, and uses that method for the child node. If there are no parent nodes with a method assigned then the application triggers a processing error stating that no assumptions are assigned for the particular product/currency combination. However, if the parent node has the "Do Not Calculate" method assigned to it then the child node inherits "Do Not Calculate", obviating the need for calculation and a processing error.

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.

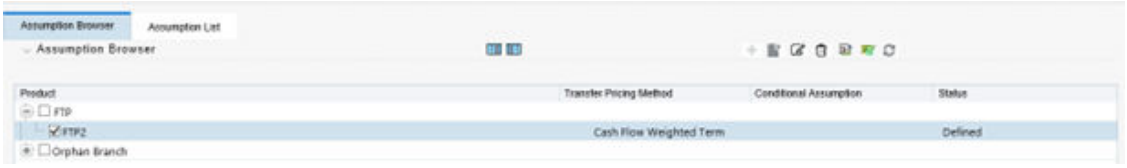
You need not specify prepayment assumptions at the same nodes as transfer pricing methods. For example, each mortgage category can have a different prepayment method while the entire Mortgage node uses the Zero Discount Factors cash flow method for transfer pricing.

11.2.18 Associating Conditional Assumptions with Assumption Rules

Oracle Funds Transfer Pricing extends the setup and maintenance of assumptions by allowing users to integrate conditional logic (optional) into the setup of transfer pricing, prepayment, and TP adjustment methods. The Caterpillar method under Transfer Pricing Rules will not be available for selection under Conditional Assumptions.

The Conditional Assumption UI is accessed from the Assumption Browser by selecting the Conditional Assumption icon.

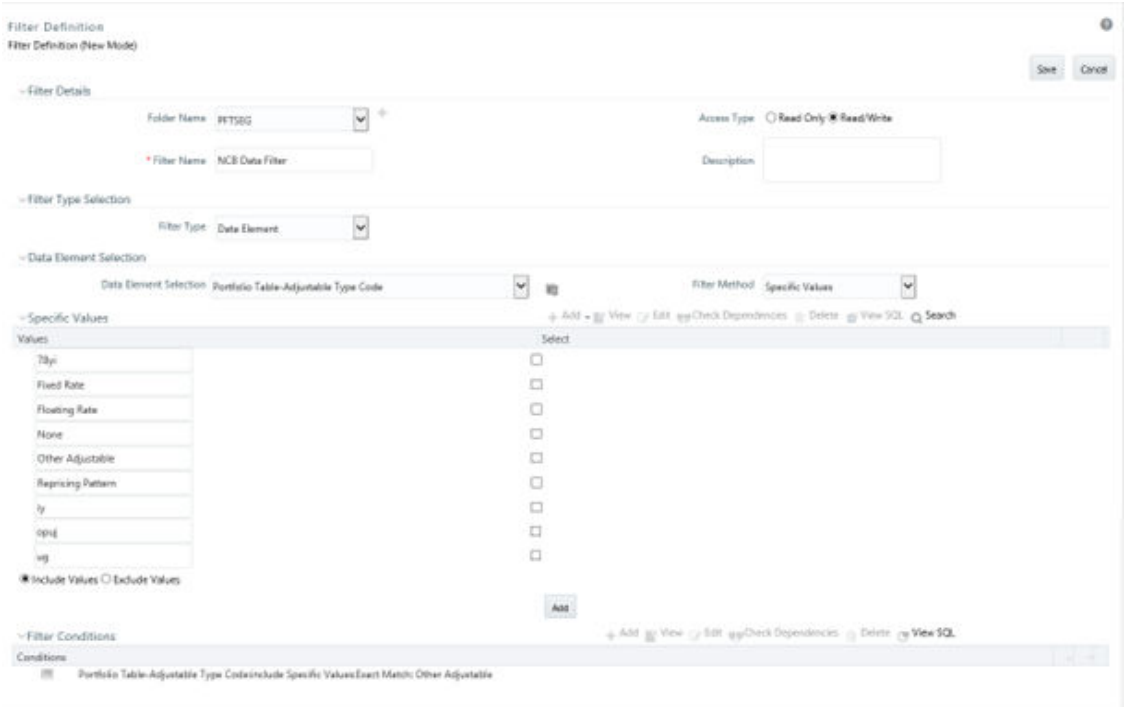
Figure 11-10 Assumption Browser



The conditional logic is defined using Data Filters and/or Maps. These existing objects provide the building blocks for defining Conditional logic. For example, each Data Filter can provide the logic for a specific condition. In the following example, the Where clause is “Adjustable Type Code = 'Adjustable Rate'”. This type of Data Filter can be selected within the Conditional Assumption UI.

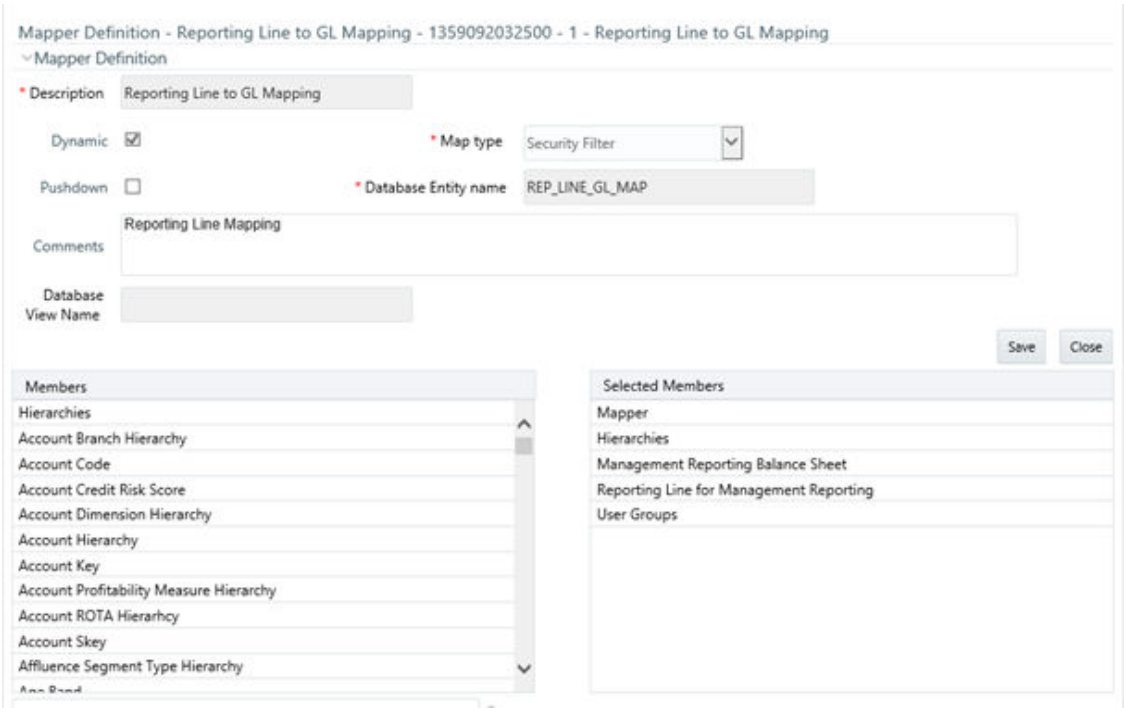
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-11 Filter Definition



Similarly, a Mapper object provides the necessary reference to one or more hierarchies, when dimension/hierarchy data is needed to define conditional logic. In the following example, this map refers to a hierarchy created on the Organization Unit dimension.

Figure 11-12 Mapper Definition- Reporting Line to GL Mapping



Copyright © 2011, 2026, Oracle and/or its affiliates.

To do the map maintenance:

1. Select **Common Object Maintenance** from the LHS menu, select **Unified Analytical Metadata**, and then select **Map Maintenance**.
2. Click the **Add** icon to create a new Map.
3. On the left-hand side, select one or more hierarchies that were enabled in the initial step.
4. Enter the required information, for example, Name, Effective Dates, and Entity Name.
5. Click **Save**.
6. From the Map Maintenance summary page, select the **Map**, and then select the **Mapper Maintenance** icon.
   Here, you will see the hierarchy and all parent/child relationships.

Columns that are part of the “Portfolio” definition determine the range of product attributes supported for conditional assumptions and available for use within Data Filters. The “Portfolio” table class is seeded with the installation and can be extended to include user-defined columns.

For more information on adding user-defined columns to the Portfolio table class, see Chapter 2 - Object Management in OFS Data Model Utilities User Guide.

When using mappers, Conditional Assumptions can be attached to any level of the hierarchy, allowing assumptions to be inherited from parent nodes by child nodes.

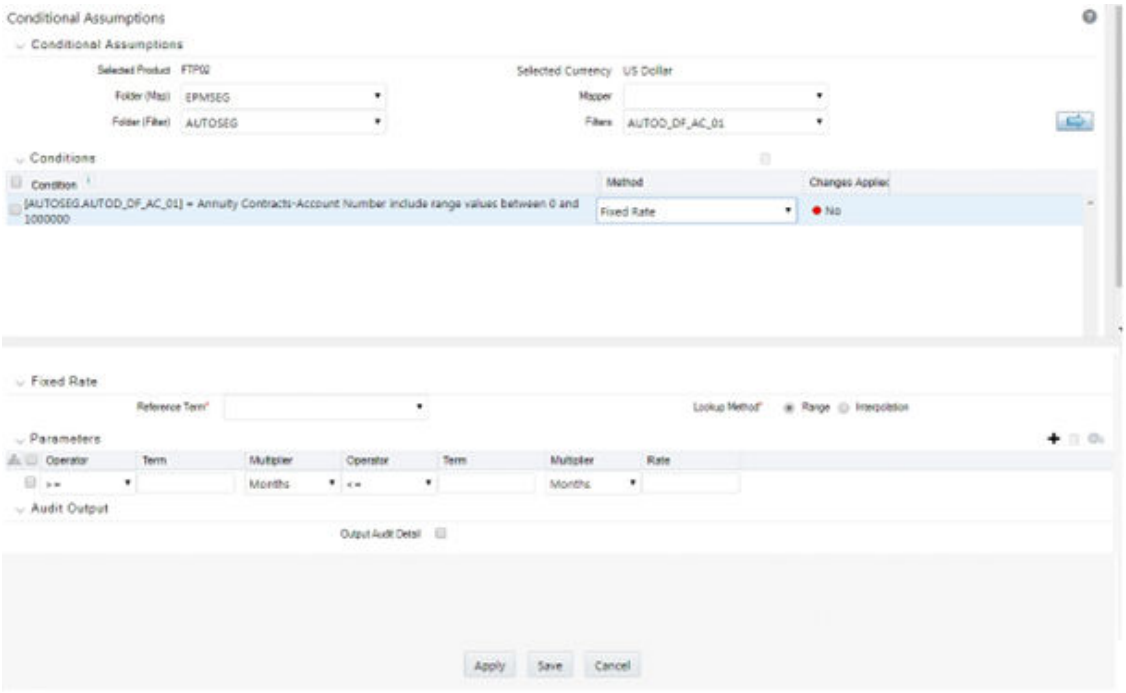
For example, you can use the Org Unit column to drive the assignment of Transfer Pricing Methods for all members of a particular Organization. You can create one Conditional Assumption to convey the entire Transfer Pricing Methodology logic and attach it to the top-level node of the Org Unit hierarchy. All nodes below the top-level node inherits the same Transfer Pricing assumption.

The logic included in a Conditional Assumption determines the specific Transfer Pricing method, Prepayment assumption, or Adjustment Rule that the system assigns to each instrument record at Run time.

The Conditional Assumption screen allows users to select explicit conditions (from Data Filters and/or Maps) and apply methods and rule selections to each condition directly. The Filter Conditions are processed by the engine in the order that they appear on the screen. After a condition is satisfied, the related assumption is applied. The following figure displays a representative Conditional Assumption using a Data Filter:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-13 Conditional Assumptions



> **Note**
>
> If an instrument record does not meet any of the conditions, then the rule logic reverts to the standard assumption that is directly assigned to the Product / Currency combination. In the example below, you can see that Fed Funds has both a direct assignment and a conditional assumption. If the condition is not met, the “Fixed Rate” assumption (ELSE condition) will be applied. For Reverse Repo's, there is only a Conditional Assumption. In the absence of an ELSE assumption, the engine will use the conditional assumption in all cases for the Product/Currency pair. To avoid this, users should define the Standard/Else assumption with appropriate input.

Conditional Assumptions can be applied only to detailed account records (data stored in the Instrument Tables). Instrument tables must have the TP - Cash Flow, Table Classification Code (200) in, or order to use Conditional Assumptions.

11.3 Define Prepayment Rules

One of the major business risks faced by financial institutions engaged in the business of lending is 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 into

Copyright © 2011, 2026, Oracle and/or its affiliates.

account while generating transfer rates. Oracle Funds 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. For more information, see Defining Transfer Pricing Methodologies.

Prepayment methodologies are associated with the product-currency combinations within the Prepayment rule. For more information, see Prepayment Rules.

Oracle Funds Transfer Pricing allows you to make use of the node level and conditional assumption while defining prepayment methodologies for your products. For more information, see Associating Node Level and Conditional Assumptions with Prepayment Rules.

> **Tip**
>
> Prepayment assumptions are used in combination with only the four cash flow based transfer pricing methods: Weighted Term, Duration, Average Life, and Zero Discount Factors.

- Prepayment Methodologies and Rules
  You can use any of the following methods in a Prepayment rule to model the prepayment behavior of instruments:
- 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.

11.3.1 Prepayment Methodologies and Rules

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

- Constant Prepayment Method
- Prepayment Model Method
- PSA Method
- Arctangent Calculation Method
- Constant Prepayment Method
  The Constant Prepayment method calculates the prepayment amount as a flat percentage of the current balance.
- Prepayment Model Method
  The Prepayment Model method allows you to define more complex prepayment assumptions compared to the other prepayment methods. Under this method, prepayment assumptions are assigned using a custom Prepayment Model.
- PSA Method
  The PSA Prepayment Method (Public Securities Association Standard Prepayment Model) is a standardized Prepayment Model that is built on a single dimension, remaining term. The PSA curve is a schedule of prepayments, which assumes that prepayments occur at a rate of 0.2 percent CPR in the first month, and increases an additional 0.2 percent CPR each month until the 30th month and prepays at a rate of 6 percent CPR thereafter ("100 percent PSA"). PSA prepayment speeds are expressed as a multiple of this base scenario. For example, a 200 percent PSA assumes annual prepayment rates will be twice as fast in each of these periods -- 0.4 percent in the first month, 0.8 percent in the second month,

Copyright © 2011, 2026, Oracle and/or its affiliates.

reaching 12 percent in month 30 and remaining at 12 percent after that. A zero percent PSA assumes no prepayments.

- 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).

11.3.1.1 Constant Prepayment Method

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

You can create your 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.

11.3.1.2 Prepayment Model Method

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

You can build a Prepayment Model 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.

A typical Prepayment Model structure includes the following:

- **Prepayment Drivers**: You can build a Prepayment Model 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.
- **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:

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 Model can contain only up to three prepayment drivers.

Prepayment drivers can be divided into the following categories:

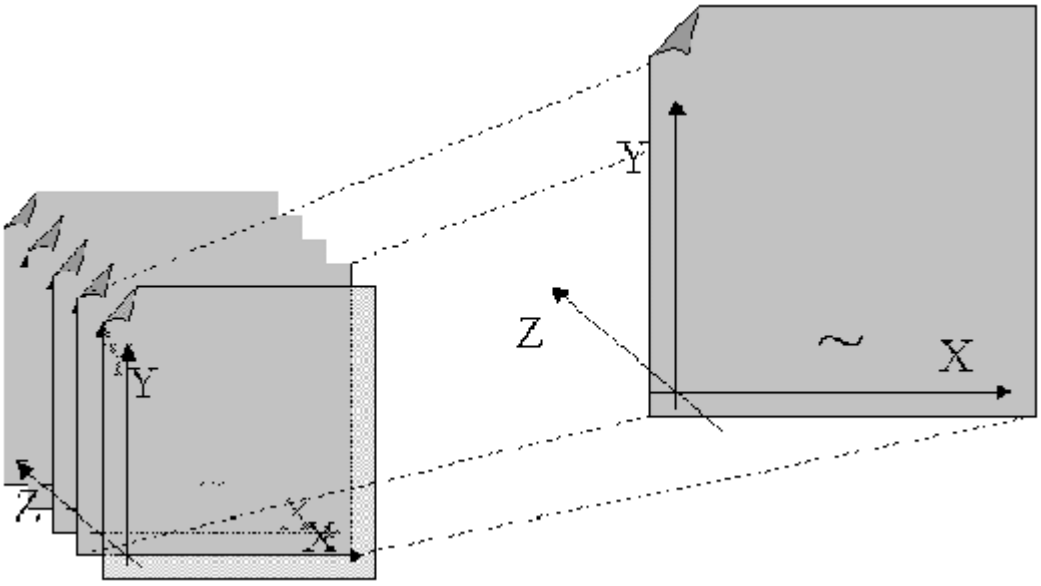
Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Age or Term Drivers**: The Age or Term drivers define term and repricing parameters in a Prepayment Model. 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 long-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 effects 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 the current gross rate to the market rate.

The following diagram illustrates a three-driver prepayment model:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-14 Three-Driver Prepayment Model



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 Funds Transfer Pricing allows you to build Prepayment Models using the Prepayment Model Rule. The Prepayment Model Rule can then be referenced by a Prepayment Rule.

11.3.1.3 PSA Method

The PSA Prepayment Method (Public Securities Association Standard Prepayment Model) is a standardized Prepayment Model that is built on a single dimension, remaining term. The PSA curve is a schedule of prepayments, which assumes that prepayments occur at a rate of 0.2 percent CPR in the first month, and increases an additional 0.2 percent CPR each month until the 30th month and prepays at a rate of 6 percent CPR thereafter ("100 percent PSA"). PSA prepayment speeds are expressed as a multiple of this base scenario. For example, a 200 percent PSA assumes annual prepayment rates will be twice as fast in each of these periods -- 0.4 percent in the first month, 0.8 percent in the second month, reaching 12 percent in month 30 and remaining at 12 percent after that. A zero percent PSA assumes no prepayments.

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

> **Note**
>
> PSA speed inputs can be between 0 and 1667.

11.3.1.4 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).

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> All prepayment rates should be input as annual amounts.

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

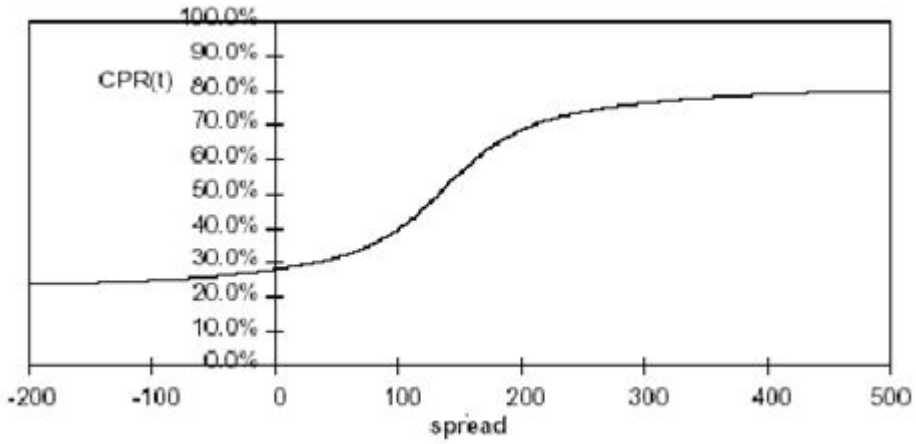
- CPRt = k1 - (k2 * ATAN(k3 * (-Ct/Mt + k4)))
- Where CPRt = annual prepayment rate in period t
- Ct = Coupon in period t
- Mt = Market rate in period t
- k1 - k4 = User-defined coefficients

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

- k1 = 0.3
- k2 = 0.2
- k3 = 10.0
- k4 = 1.2

Each coefficient affects the prepayment curve in a different manner.

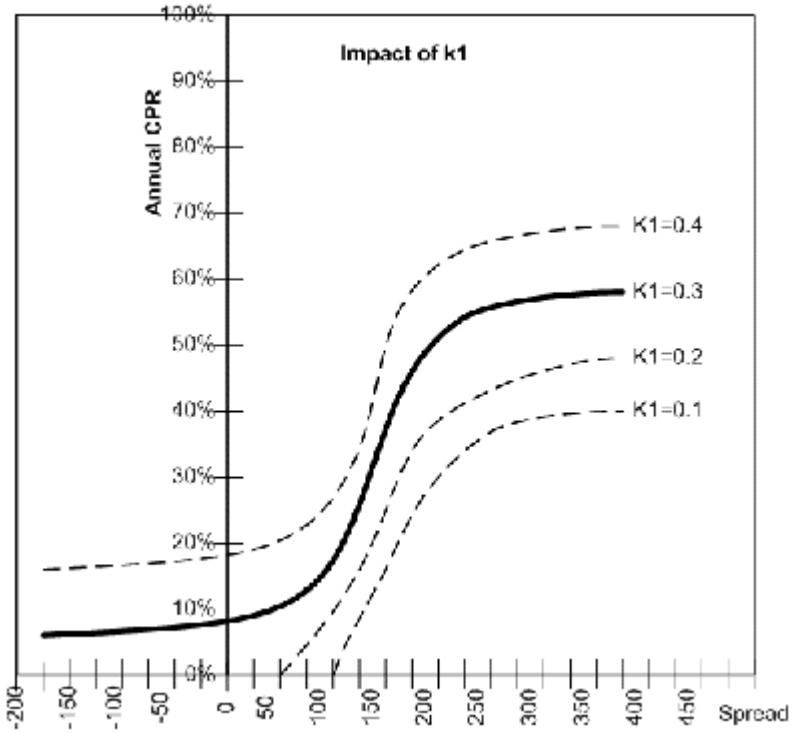
Figure 11-15 Prepayment Curve



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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

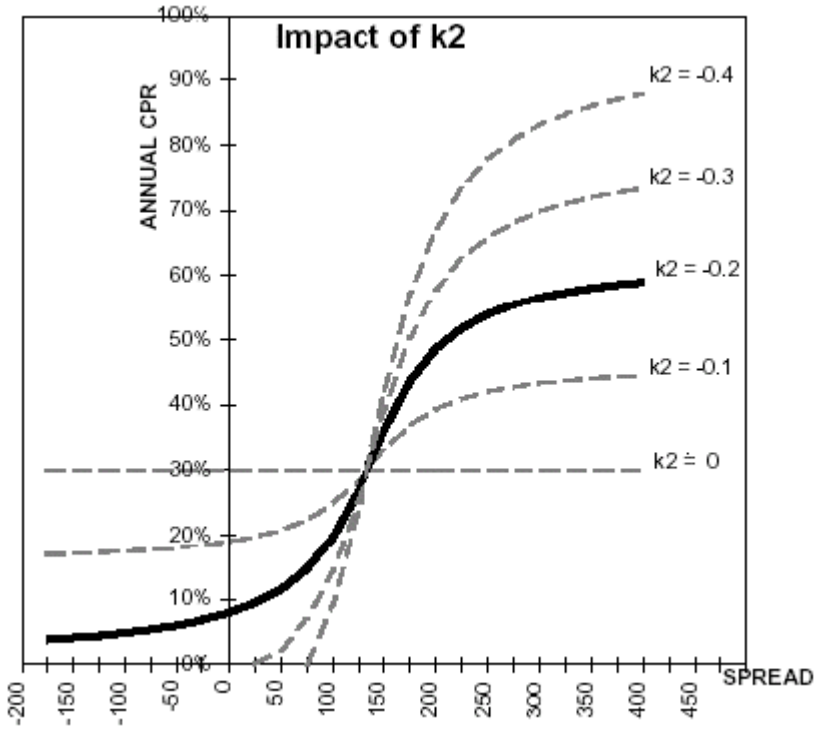
Figure 11-16 Impact of K1 on the Prepayment Curve



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.

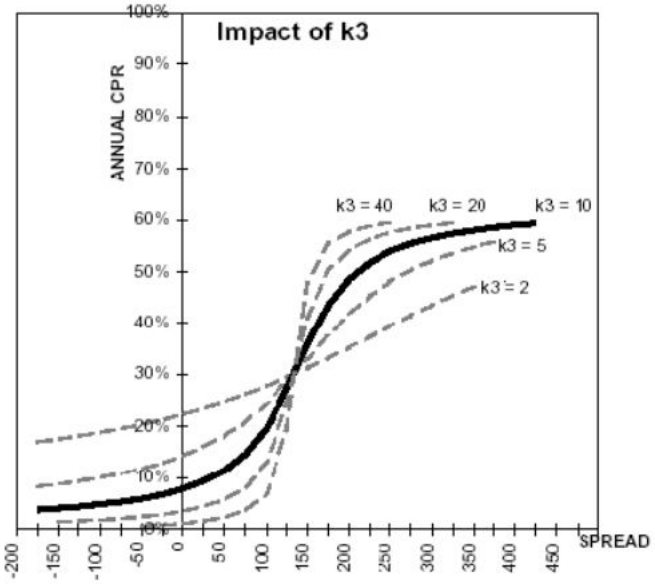
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-17 Impact of K2 on the Prepayment Curve



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.

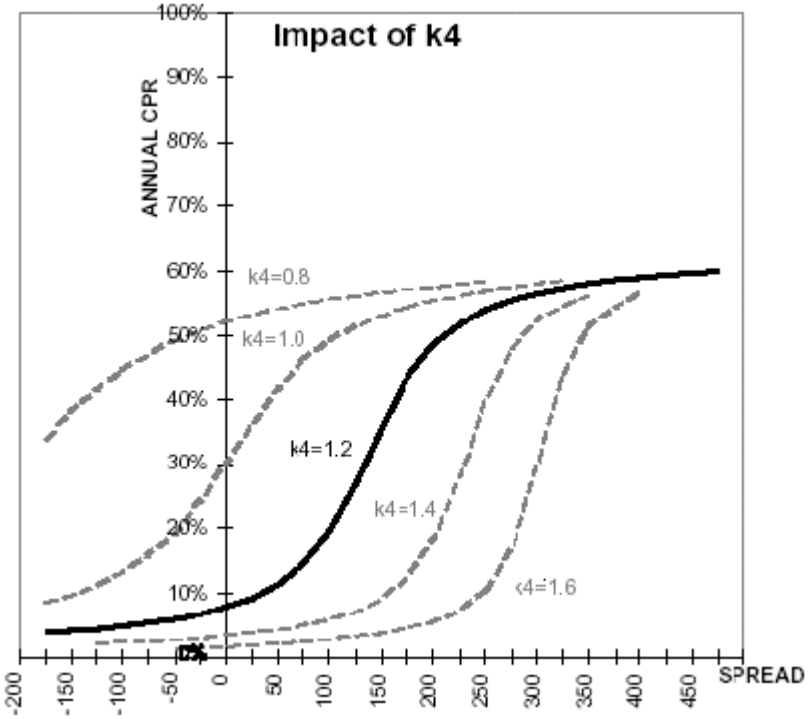
Figure 11-18 Impact of K3 on the Prepayment Curve



Copyright © 2011, 2026, Oracle and/or its affiliates.

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.

Figure 11-19 Impact of K4 on the Prepayment Curve



11.3.2 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. For more information, see Defining Transfer Pricing Methodologies Using Node Level Assumptions.

You can also use the conditional logic of Conditional Assumptions to define prepayment methodologies based on particular characteristics of financial instruments. For more information, see Associating Conditional Assumptions with Assumption Rules.

11.4 Define Adjustment Rules

Adjustment Rules allow users to define Transfer Pricing Add-on rates that are assigned incrementally to the base FTP rate to account for a variety of miscellaneous risks such as Liquidity risk or Basis risk or to supplement strategic decision-making with Pricing Incentives, Breakage Charges, or other types of rate adjustments.

Within both the Standard and Stochastic Transfer Pricing Processes, users can select an appropriate Adjustment rule to calculate add-on rates or breakage charges.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Add-on rates can be a fixed rate, a fixed amount, or a formula based rate. Breakage Charges can be a fixed percentage, a fixed amount or can be calculated on an Economic Loss basis. The adjustments are calculated and output separately from the base funds transfer pricing rate, so they can be easily identified and reported. Additionally, adjustments allow you to apply event-based logic through the use of conditional assumptions that are applied or varied only if a specific condition is satisfied.

- Adjustment Methods and Rules
  You can use any of the following methods in an Adjustment Rule when the selected Adjustment Type is Liquidity Premium, Basis Risk Cost, Pricing Incentive, or Other Adjustment:
- Breakage Charges
  A Breakage Charge represents the cost of breaking a contractual obligation. In Bank Finance this means the early prepayment of a loan by a customer or the early withdrawal of deposit funds by a customer. “Early” in this sense means before the contractual maturity date.

11.4.1 Adjustment Methods and Rules

You can use any of the following methods in an Adjustment Rule when the selected Adjustment Type is Liquidity Premium, Basis Risk Cost, Pricing Incentive, or Other Adjustment:

- Adjustment - Fixed Rate
- Adjustment - Fixed Amount
- Adjustment - Formula Based Rate
- Adjustment - Use TP Method from selected TP Rule

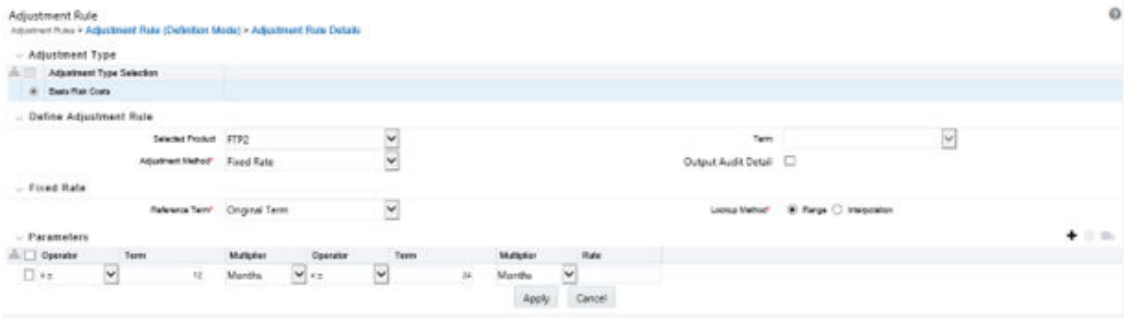
Alternatively, you can use any of the following methods in an Adjustment Rule when the selected Adjustment Type is Breakage Charge:

- Economic Loss
- Fixed Amount
- Fixed Percentage
- Adjustment - Fixed Rate
- Adjustment - Fixed Amount
- Adjustment - Formula Based Rate
- Adjustment - Use TP Method from Selected TP Rule
  The Use TP Method from Selected TP Rule selection allows the user to calculate the add-on rate based on any TP method available in the selected Transfer Pricing Rule.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.4.1.1 Adjustment - Fixed Rate

Figure 11-20 Adjustment Rule Details



The Fixed Amount Adjustment method allows the user to associate an amount with specific terms or term ranges. Reference term selections include the following:

- **Repricing Frequency**: The fixed amount is matched to the specified reprice frequency of the instrument. If the instrument is a fixed rate and, therefore, does not have a reprice frequency, the fixed amount lookup happens based on the original term of the instrument.
- **Original Term**: The calculation assigns the fixed amount based on the original term on the instrument.
- **Remaining Term**: The calculation assigns the fixed amount based on the remaining term of the instrument.

The remaining term value represents the remaining term of the contract and is expressed in days.

- **Holiday Calendar**: Select if a holiday calendar is applicable for calculating the charges/ credits.
- **Rolling Convention**: Select the appropriate business day rolling convention if a Holiday Calendar is selected.
- **Interest Calculation logic**: Select the appropriate option to indicate how the interest payment should be adjusted when a holiday date is encountered.

Remaining Term = Maturity Date – As-of-Date.

- **Duration (read from the TP_DURATION column)**: The calculation assigns the fixed amount based on the Duration of the instrument, specified in the TP_DURATION column.
- **Average Life (read from the TP_AVERAGE_LIFE column)**: The calculation assigns the fixed amount based on the Average Life of the instrument, specified in the TP_AVG_LIFE column.

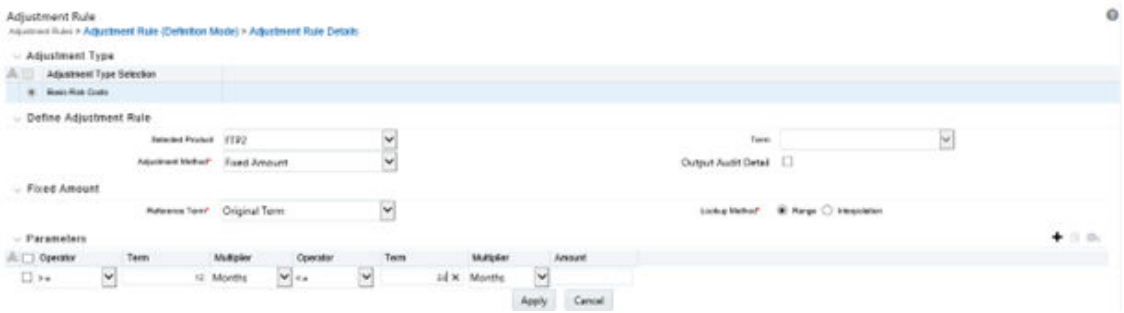
You can create your reference term ranges and assign a particular adjustment amount to all instruments with a reference term falling within the specified range.

> **Note**
>
> All adjustment rates should be input as annual rates.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.4.1.2 Adjustment - Fixed Amount

Figure 11-21 Adjustment Rule Details



The Fixed Amount Adjustment method allows the user to associate an amount with specific terms or term ranges. Reference term selections include the following:

- **Repricing Frequency**: The calculation retrieves the rate for the term point equaling the reprice frequency of the instrument. If the instrument is a fixed rate and, therefore, does not have a reprice frequency, the calculation retrieves the rate associated with the term point equaling the original term on the instrument.
- **Original Term**: The calculation retrieves the rate for the term point equaling the original term on the instrument.
- **Remaining Term**: The calculation retrieves the rate for the term point corresponding to the remaining term of the instrument. The remaining term value represents the remaining term of the contract and is expressed in days. Remaining Term = Maturity Date – As-of-Date.
- **Duration (read from the TP_DURATION column)**: The calculation retrieves the rate for the term point corresponding to the Duration of the instrument, specified in the TP_DURATION column.
- **Average Life (read from the TP_AVERAGE_LIFE column)**: The calculation retrieves the rate for the term point corresponding to the Average Life of the instrument, specified in the TP_AVG_LIFE column.

You can create your reference term ranges and assign a particular adjustment amount to all instruments with a reference term falling within the specified range.

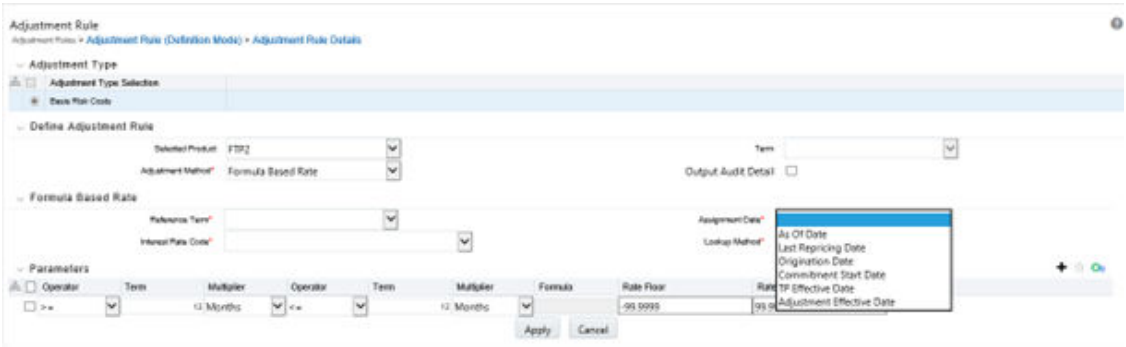
- **Holiday Calendar**: Select if a holiday calendar is applicable for calculating the charges/ credits.
- **Rolling Convention**: Select the appropriate business day rolling convention if a Holiday Calendar is selected.
- **Interest Calculation logic**: Select the appropriate option to indicate how the interest payment should be adjusted when a holiday date is encountered.

> **Note**
>
> All adjustment amounts should be input in base currency for the selected product/ currency combination.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.4.1.3 Adjustment - Formula Based Rate

Figure 11-22 Adjustment Rule Details - Formula Based Rate



The Formula Based Rate Adjustment method allows the user to determine the add-on rate based on a lookup from the selected yield curve, plus a spread amount, and then the resulting rate can be associated with specific terms or term ranges. Reference term selections include:

- **Repricing Frequency**: The calculation retrieves the rate for the term point equaling the reprice frequency of the instrument. If the instrument is a fixed rate and, therefore, does not have a reprice frequency, the calculation retrieves the rate associated with the term point equaling the original term on the instrument.
- **Original Term**: The calculation retrieves the rate for the term point equaling the original term on the instrument.
- **Remaining Term**: The calculation retrieves the rate for the term point corresponding to the remaining term of the instrument. The remaining term value represents the remaining term of the contract and is expressed in days.
  Remaining Term = Maturity Date – As-of-Date
- **Duration (read from the TP_DURATION column)**: The calculation retrieves the rate for the term point corresponding to the Duration of the instrument, specified in the TP_DURATION column.
- **Average Life (read from the TP_AVERAGE_LIFE column)**: The calculation retrieves the rate for the term point corresponding to the Average Life of the instrument, specified in the TP_AVG_LIFE column.

You can create your reference term ranges and assign a particular formula based adjustment rate to all instruments with a reference term falling within the specified range.

With this method, you also specify the Interest Rate Code and define an Assignment Date for the Rate Lookup. The Interest Rate Code can be any IRC defined within Rate Management, but will commonly be a Hybrid IRC defined as a Spread Curve (for example, Curve A – Curve B).

Assignment Date selections include:

- As-of-Date
- Last Repricing Date
- Origination Date
- TP Effective Date

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Adjustment Effective Date
- Commitment Start Date

Each term range additionally allows users to input a Rate Cap and a Rate Floor. These boundaries will only apply to the 'Formula Based Method' and 'Use TP Method from TP Rule' based adjustments. These are optional inputs. This input limits the Max or Min rate regardless of the rate passed by the Formula/TP Rule. Sometimes, there may be major external events that cause a short-term spike in rates and certain accounts may be negatively impacted as a result. Applying a rate cap (or floor) allows business users to limit these spikes.

> **Note**
>
> Term range considers one month equal to 30.416667 days and 1 year = 365 days, therefore, 12 Months would marginally be more than one year by 0.000004 days.

The formula definition is comprised of the following components.

Figure 11-23 Adjustment Rate Details - Formula Based Rates



- **Term Point**: This allows you to associate a specific term point from the IRC to each Term Range.
- **Coefficient**: This allows you to define a multiplier that is applied to the selected rate.
- **Rate Spread**: This allows you to define an incremental rate spread to be included on top of the IRC rate.

The resulting formula for the adjustment rate: (Term Point Rate * Coefficient) + Spread For example:

Figure 11-24 Formula Rate Parameters



> **Note**
>
> For increased precision, you can reduce the Term Ranges to smaller term increments allowing you to associate specific IRC rate tenors with specific terms.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Holiday Calendar**: Select if a holiday calendar is applicable for calculating the charges/ credits.
- **Rolling Convention**: Select the appropriate business day rolling convention if a Holiday Calendar is selected.
- **Interest Calculation logic**: Select the appropriate option to indicate how the interest payment should be adjusted when a holiday date is encountered.

11.4.1.4 Adjustment - Use TP Method from Selected TP Rule

The Use TP Method from Selected TP Rule selection allows the user to calculate the add-on rate based on any TP method available in the selected Transfer Pricing Rule.

Figure 11-25 Adjustment Rule (Edit Mode)

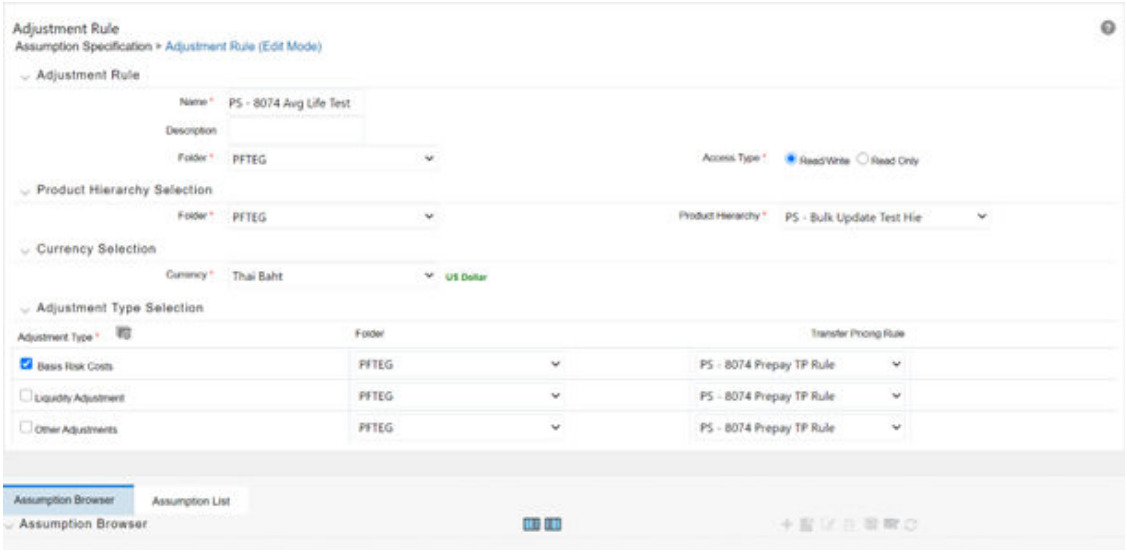
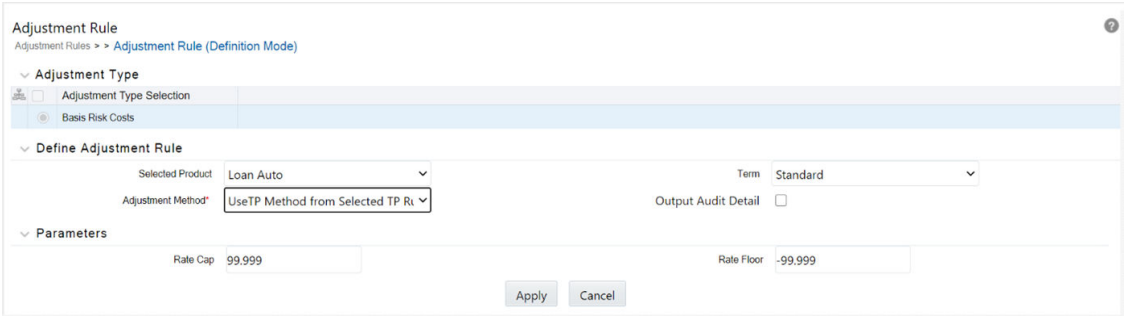


Figure 11-26 Adjustment Rule Details - TP Method



Users can attach any Transfer Pricing Rule on the Adjustment Rule summary page separately for each type of add-on rate. The TP Methods mapped to product hierarchy members in the TP Rule will be read during the adjustment rate calculation process and applied during the calculation of the Adjustment Rate(s). Outputs will be written to the respective Adjustment Rate column, for example, Basis Risk Cost Rate, Liquidity Premium Rate, Pricing Incentive Rate, or Other Adjustment Rate.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Term**:
  - **Standard Term**: Adjustment would be calculated as per the repricing period for adjustable-rate instruments and use the original term (maturity date - origination date) for fixed-rate instruments.
  - **Original Term**: Adjustments would be calculated as per the original term like a fixed-rate instrument.

If the underlying TP method is (Cash Flow: Average Life or Cash Flow: Duration) and output average life or duration is selected.

A table FSI_O_FTP_ADJ_TP_VALUES has been introduced to store average life or duration values for each of the adjustment types. Each value will be tagged to a particular adjustment type based on the following ADJUSTMENT_TYPE_CD's:

- Liquidity Adjustment (1)
- Basis Risk Cost (2)
- Pricing Incentive (3)
- Other Adjustments (4)

The same account can have multiple average life/duration values stored against a particular adjustment type; as values are not overwritten and can be uniquely identified using Batch Run ID.

Adjustments using cash flow TP methods will generate cash flows and be stored in the FSI_O_PROCESS_CASH_FLOWS table, irrespective of whether TP Average Life or TP Duration is updated in the instrument record or not.

Additionally, a Delete Batch procedure is introduced – This purge batch has been provided to delete records in FSI_O_FTP_ADJ_TP_VALUES as per selected from and to "As-of-Date".

> **Note**
>
> Adjustment Rates are always processed row by row rather than in bulk, so care should be taken when selecting a TP Method for use in calculating an Adjustment Rate. Specifically, bulk methods should be tested to ensure results and performance are as expected.

- Setting up Purge Batch for the FSI_O_FTP_ADJ_TP_VALUES Table
  To delete records containing average life/duration for adjustment rate calculations from the FSI_O_FTP_ADJ_TP_VALUES table.

11.4.1.4.1 Setting up Purge Batch for the FSI_O_FTP_ADJ_TP_VALUES Table

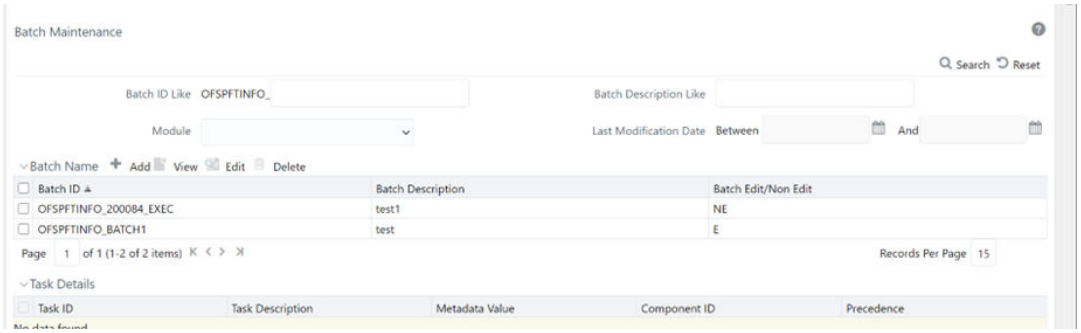
To delete records containing average life/duration for adjustment rate calculations from the FSI_O_FTP_ADJ_TP_VALUES table.

To create a Batch:

1. Click the Batch Maintenance option under Operations.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-27 Batch Maintenance page

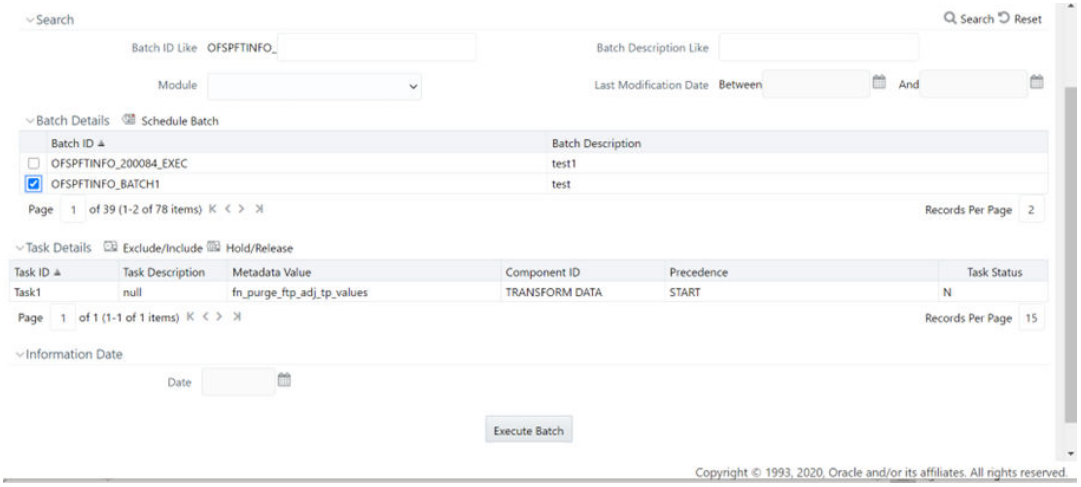


2. Click the **Add** icon to create a Batch for deleting records from the FSI_O_FTP_ADJ_TP_VALUES table.
3. Enter the **Batch Name** and **Description**. Select the Duplicate Batch and Sequential Batch, or Duplicate Batch or Sequential Batch options if required. Click **Save**.
4. Search the created Batch using the Search criteria if required. The Batch is displayed. Select the Batch to add the corresponding task to it.
5. Click **Add** to add the task to the selected Batch, enter the description Task Definition page. Select the component of the task as Transform Data.

   Enter the required parameter details as follows:

   - Datastore Type: Leave the default value of “EDW”.
   - Datastore Name: Leave the default value.
   - IP Address: Leave the default value.
   - Rule Name: Purge ADJ tp values.
   - Parameter List: (Date Range): ‘From Date’, ‘To Date’ Format('YYYYMMDD').
6. To execute the Batch, click the **Batch Execution** option under **Operations**. Select the Batch, which you want to execute using the search criteria. Select the Batch using the corresponding check box.

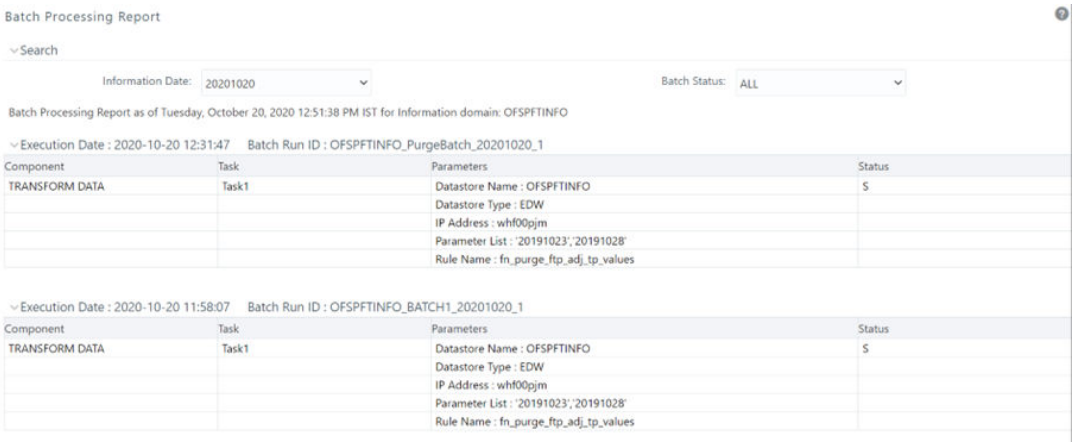
Figure 11-28 Batch Search page



Copyright © 2011, 2026, Oracle and/or its affiliates.

7. You can also use the Schedule option if you want to schedule a Batch for later or recurring processing. Click **Execute Batch**. A message is displayed: *Batch Triggered Successfully*. Click **Ok**.
8. To view the Batch Processing Report, click the **Batch Processing Report** option under. Expand the Execution Date.

Figure 11-29 Batch Processing Report



11.4.2 Breakage Charges

A Breakage Charge represents the cost of breaking a contractual obligation. In Bank Finance this means the early prepayment of a loan by a customer or the early withdrawal of deposit funds by a customer. “Early” in this sense means before the contractual maturity date.

The gain or loss to the Bank from such early prepayments and withdrawals is the opportunity cost of not being able to replace the spread earned on the asset or deposit being lost. For example, the early withdrawal of funds from a 2-year term deposit exposes the bank to the risk of replacing that funding in a higher rate environment and thereby reducing the net interest margin earned before the withdrawal. With matched-term transfer pricing, this risk is split between the Line Unit and Treasury. The Line Unit holds the risk of deteriorating credit spread, but Treasury holds the funding risk (the risk that the funding spread between the Bank's assets and liabilities will narrow).

The following Breakage Charge methods are available including:

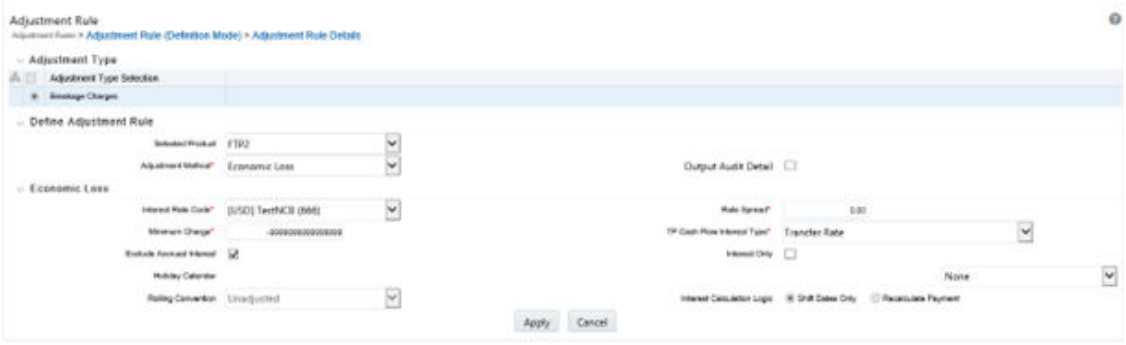
- Economic Loss
- Fixed Amount
- Fixed Percentage

- Breakage Charge – Economic Loss
- Breakage Charge – Fixed Amount
- Breakage Charge - Fixed Percentage
- Prerequisites for calculating Breakage Charges
  The following are the prerequisites for calculating the Breakage Charges:
- Associating Node Level and Conditional Assumptions with Adjustment Rules
  You can define adjustment methods at any level of the product hierarchy. Children of a hierarchical node automatically inherit the assumptions defined at the parent level.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.4.2.1 Breakage Charge – Economic Loss

Figure 11-30 Adjustment Rule Details - Adjustment Method as Economic Loss



The Economic Loss Breakage Charge method sets out to compute the cost to the organization (economic loss) incurred for terminating the funding liability (also known as the shadow liability). The calculation assumes the funding liability has the exact attributes of the funded/ terminated instrument. The rate of the funding liability is equal to the Transfer Rate. Economic Loss is computed as follows:

For Assets:

Economic Loss =BV - MV

For Liabilities: Economic Loss = MV– BV

Where:

- MV: Market Value of the funding Liability
- BV: Book Value of the broken instrument

The following is a simplified example of the Economic Loss calculation for a standard Term Deposit:

- Book Value: $1,000.00
- Original Term: 24 Months
- Break after: 12 Months
- Original TP Rate: 2.40% (based on straight term method)

Table 11-11 Reference Rates

Effective Date	1 M	12 M	24 M
At Origination	2.00	2.40	1.75
At Month 12	2.00	2.40	1.75

Table 11-12 Cash flows of remaining funding after break event

		Original TP COF	Total

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-12 (Cont.) Cash flows of remaining funding after break event

Month	Principal	@ 2.40%	CF Orig TP
13		$ 2.00	$2.00
14		$2.00	$2.00
15		$2.00	$2.00
16		$2.00	$2.00
17		$2.00	$2.00
18		$2.00	$2.00
19		$2.00	$2.00
20		$2.00	$2.00
21		$2.00	$2.00
22		$2.00	$2.00
23		$2.00	$2.00
24	$1,000.00	$2.00	$1,002.00
Market Value at Month 12		1,003.957	
Book value		-1,000.00	
Breakage charge		3.957	

Note

If you are calculating Breakage Charges, using the Economic Loss method, you must select the "Remaining Term" option in your Transfer Pricing Process, to generate the correct cash flows for the funding liability.

Both Current and Prior period cash flows will be logged which are used for Break Charge calculations in FSI_O_PROCESS_CASH_FLOWS, table using the Record Sequence (n,n+1) for Prior/Current period record respectively. Thus, the ID Number and Record Sequence can help to identify if Cash Flows in the FSI_O_PROCESS_CASH_FLOWS belong to prior or current period record.

Note

System calculates breakage cost based on the difference between market value and book value. In a few boundary scenarios, like accounts where the existing transfer rate is close to the current market rate, or where the instrument has a single bullet payment at maturity, this method may indicate a gain where an economic loss is expected, or a loss where an economic gain is expected.

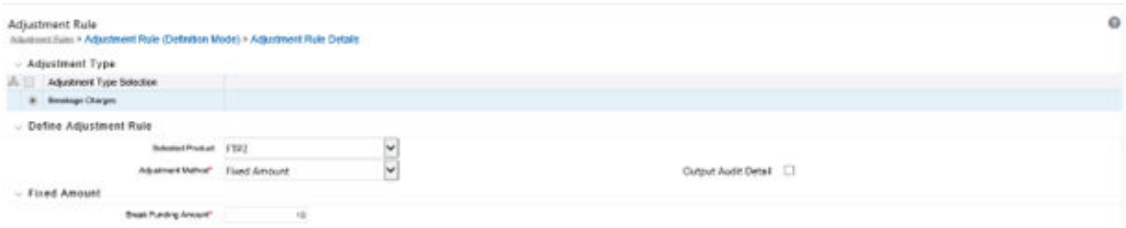
To handle such anomalies, the system directly calculates breakage cost from the rate spread (Difference of existing Transfer and Market Rate) and the discounted value of the spread-based cash flows is recorded as the Break Fund Amount/Break Fund MV.

The `INCLUDE_SPREAD_BASED_BREAKAGE_COST_CALC` parameter controls the spread-based calculation used to handle these anomalies. The default value in the drop-down list is **NO**. Select **YES** to enable the spread-based approach.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.4.2.2 Breakage Charge – Fixed Amount

Figure 11-31 Adjustment Rule Details - Adjustment Method as Fixed Amount



The Fixed Amount method allows users to directly input the amount of the Breakage Charge. This method would be used in cases where the cash flows and the Economic Loss method are not appropriate for determining the breakage cost.

The only input required for this method is the Breakage Charge amount.

11.4.2.3 Breakage Charge - Fixed Percentage

Figure 11-32 Adjustment Rule Details - Adjustment Method as Fixed Percentage



An alternative to the Fixed Amount method, the Fixed Percentage approach allows you to input a percentage that is multiplied by the breakage amount to determine the Breakage Charge.

Calculation:

```
Breakage Charge = Break Amount x (Charge % / 100)
```

If the resulting amount is greater than the specified minimum charge, the calculated amount is output. Otherwise, the minimum charge will be output.

11.4.2.4 Prerequisites for calculating Breakage Charges

The following are the prerequisites for calculating the Breakage Charges:

1. Break event records should be populated in the FSI_D_BREAK_FUNDING_CHARGES table. This can be done directly through a manual data loading process or can be automated using an FTP Break Identification Process (see Break Identification Process).
2. An Adjustment Rule should be defined with Breakage Charge assumptions created for all of the relevant Product / Currency combinations (see Adjustment Rules).

Copyright © 2011, 2026, Oracle and/or its affiliates.

3. A Standard TP Process should be defined with the following: (see Transfer Pricing Process).
   - The Break Funding Charges table selected on the Product Selection block.
   - Adjustments must be selected on the Calculation Elements block.
   - An Adjustment Rule containing the required Breakage Charge assumptions must be selected.

11.4.2.5 Associating Node Level and Conditional Assumptions with Adjustment Rules

You can define adjustment methods 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. For more information, see Defining Transfer Pricing Methodologies Using Node Level Assumptions.

You can also use the conditional logic of Conditional Assumptions to define Adjustment methods based on the particular characteristics of your financial instruments. For more information, see Associating Node Level and Conditional Assumptions with Prepayment Rules.

11.5 Define Alternate Rate Output Mapping Rules

Within the Standard Transfer Pricing Process, users can select an appropriate Alternate Rate Output Mapping rule to output transfer pricing results to the seeded or user-defined alternate columns instead of the standard default column.

The Alternate Rate Output Mapping Rule lets you select the alternate columns to output transfer rate, option cost, and adjustment calculation results for each instrument record for a Transfer Pricing Process Run. This functionality allows you to output more than one transfer rate, option cost, or adjustment calculation result for each instrument record through Multiple Transfer Pricing Process Runs.

For example, you may Run Oracle Funds Transfer Pricing once without selecting an Alternate Rate Output Mapping rule and thus use the original default output columns, such as TRANSFER_RATE and MATCHED_SPREAD_C, to store transfer pricing results. You may then run the application a second or third time using an Alternate Rate Output Mapping rule in different Transfer Pricing Processes, which Run against the same instrument records. The result would be two or three transfer rates, option costs, or adjustment calculation results populated into distinct columns for each instrument record in the specified Instrument tables.

11.6 Define and Execute a Transfer Pricing Process

Defining and executing a Transfer Pricing Process is one of the mandatory steps in Oracle Funds Transfer Pricing. The Transfer Pricing Process allows you to:

- Select Transfer Pricing, Prepayment, and Adjustment Assumptions, respectively, contained in the associated Transfer Pricing, Prepayment, and Adjustment rules, within the Transfer Pricing Process.
- Determine the data that you want to process in a particular Run.
- Define the calculation elements used in Transfer Rate, Option Cost, Economic Value, and Adjustment Calculations to migrate transfer pricing results to the Management Ledger table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Choose the calculation mode for generating transfer pricing results, such as the Remaining Term or Standard.
- Select which calculations, such as Transfer Rates, TP Rate Adjustments, or Option Costs to perform.
- Optionally select an Alternate Rate Output Mapping Rule. Alternate Rate Output Mapping rules can be used when you want to direct the FTP output to non-standard columns.

After a Transfer Pricing Process has been created, a user may select RUN from the summary page to execute it.

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

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

Table 11-13 Instrument Table Fields Updated by Transfer Pricing Processing

Calculation Type	Calculation Mode	Account Table Field
Transfer Rates	Standard	TRANSFER_RATE and MATCHED_SPREAD_C
CHARGE_CREDIT_TRATE		
Transfer Rates	Remaining Term	TRAN_RATE_REM_TERM
CHARGE_CREDIT_TRATE_REM_TERM	Standard	HISTORIC_STATIC_SPREAD and HISTORIC_OAS
Option Costs		
Option Costs	Remaining Term	CUR_STATIC_SPREAD and CUR_OAS
Adjustments - Liquidity Adjustment	Any	LIQUIDITY_PREMIUM_RATE
LIQUIDITY_PREMIUM_AMT		
LIQUIDITY_PREM_CHARGE_CREDIT		
Adjustments - Basis Risk Cost	Any	BASIS_RISK_COST_RATE
BASIS_RISK_COST_AMT		
BASIS_RISK_CHARGE_CREDIT		
Adjustments - Pricing Incentives	Any	PRICING_INCENTIVE_RATE
PRICING_INCENTIVE_AMT		
PRICING_INC_CHARGE_CREDIT		
Adjustments - Other Adjustments	Any	OTHER_ADJUSTMENTS_RAT
OTHER_ADJUSTMENTS_AMT		
OTHER_ADJ_CHARGE_CREDIT		
Adjustments - Breakage Charges	Remaining Term	BREAK_FUNDING_RATE
BREAK_FUNDING_AMT		
BREAK_FUNDING_MV		
Economic Value- All In TP Rate	Any	EV_ALLIN_TP_RATE
Economic Value - TP Rate	Any	EV_TP_RATE
Economic Value - Liquidity Premium Rate	Any	EV_LIQ_PREM_RATE

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-13 (Cont.) Instrument Table Fields Updated by Transfer Pricing Processing

Calculation Type	Calculation Mode	Account Table Field
Economic Value - Pricing Incentive Rate	Any	EV_PRIC_INC_RATE
Economic Value - Basis Risk Rate	Any	EV_BASIS_RISK_RATE
Economic Value - Other Adjustment Rate	Any	EV_OTH_ADJ_RATE
Economic Value - Other Adjustment Rate	Alt	EV_OTH_ADJ_RATE_ALT

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

- Financial Element 170, Average Transfer Rate
- Financial Element 450, Charge/Credit
- Financial Element 172, Average Remaining Term Transfer Rate
- Financial Element 452, Charge/Credit Remaining Term

> **Note**
>
> For a given combination of Organizational Unit and Product dimensions, only one row should exist for transfer rate (170 or 172) and charge/credits (450 or 452).

> **Tip**
>
> Depending on the Charge Credit Balance type selected in your Application Preferences, Either Average Balance, Financial Element 140, or Ending Balance, Financial Element 100, must exist in the Ledger Table to successfully transfer price ledger balances.

TP_EFFECTIVE_DATE and ADJUSTMENT_EFFECTIVE_DATE are used by the FTP Standard Process as an override date for determining the historical rate from the specified interest rate curve (IRC). TP_EFFECTIVE_DATE and ADJUSTMENT_EFFECTIVE_DATE are used in cases where the customer rate on the instrument record is set before the origination date and users wish to align the TP Rate lookup with the date that the customer rate was set.

The FTP process normally uses the ORIGINATION_DATE for fixed-rate instruments or LAST_REPRICE_DATE for adjustable-rate instruments to determine the historical rate effective date for the transfer rate or Adjustment Rate lookup. This input allows users to specify an alternate effective date.

If applicable, set the TP_EFFECTIVE_DATE = ISSUE_DATE or the date on which the customer rate was set.

- TP_EFFECTIVE_DATE should be NULL if not applicable. Do not default TP_EFFECTIVE_DATE (for example, 01-JAN-1900) to a valid date as the TP Engine will reference any valid date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- TP_EFFECTIVE_DATE <> ORIGINATION_DATE
- TP_EFFECTIVE_DATE <> LAST_REPRICE_DATE
- Transfer Pricing Process and the Propagation Pattern
  Transfer Pricing theory suggests that a Transfer Rate on a fixed-rate instrument should apply throughout its entire life (for Fixed Rate Instruments) or Repricing Term (for Adjustable Rate Instruments). We apply this theory to Transfer Rate propagation.
- Transfer Pricing Process and Audit Options
  Transfer Pricing Processes provide you with three audit options: Detailed Cash Flow, Forward Rates, and 1 Month Rates. While the Detailed Cash Flow audit option applies to both Standard and Stochastic TP Processes, the Forward Rates and 1 Month Rates audit options are applicable only to Stochastic TP processing. From Release 8.1.1, FTP Audit will also capture the prepayment method used, IRC, Model along with other details like TP method, adjustment method, and so on.
- SQL Query Results
  The results of running a SQL query can be exported to an Excel file for validation.
- Data Verification
  You can export results from your SQL query into Excel format and use Excel to validate the results. If the cash flows do not behave as expected, examine instrument table data or your assumptions.
- Transfer Pricing Process Rule and Calculation Modes
  You can choose to transfer price of your product portfolio either in the Standard or in Remaining Term calculation mode.
- Transfer Pricing Process and Migration Options
  The purpose of the Ledger Migration process is to generate dollar charges or credits for funds provided or used for a combination of dimensions. The information necessary to generate these charges or credits (through transfer rates, adjustments, and option cost processing) originates from the Instrument Tables and the results are inserted into the Management Ledger table and are available for use in the calculation of profitability and risk measures. Within the Transfer Pricing Process definition screen, on the "Migration" block, users can select from two options for the charge/credit method:
- Transfer Pricing Process and Calculating Economic Value
  You can choose to calculate Economic Value as part of a Standard Transfer Pricing Process by selecting the Economic Value calculation element. This calculation option refers to the Economic Value assumptions defined within the Transfer Pricing Rule and is also eligible for Alternate Rate Output mapping. Additionally, there are seeded output columns available corresponding to each of the seeded interest type elements.
- Stochastic Transfer Pricing Process and Option Cost Parameters
  In addition to transfer rates (created by a Standard TP Process), the Stochastic Transfer Pricing Process allows you to calculate the cost of options that are associated with your instruments. If you want to calculate option costs, you need to define the parameters used in option costing within the Stochastic Transfer Pricing Process.
- Stochastic Transfer Pricing Process and Stochastic Rate Indexing Rules
  The Stochastic Rate Indexing rule is one of the assumption rules that you need to select within a Stochastic Transfer Pricing Process to calculate option costs.

11.6.1 Transfer Pricing Process and the Propagation Pattern

Transfer Pricing theory suggests that a Transfer Rate on a fixed-rate instrument should apply throughout its entire life (for Fixed Rate Instruments) or Repricing Term (for Adjustable Rate Instruments). We apply this theory to Transfer Rate propagation.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> Adjustment Rates will be propagated regardless of their fixed or adjustable-rate characteristic.

The Propagation Patterns allow you to pull forward (propagate) the Transfer Rate and Matched Spread on any applicable instrument record from a prior period of history. Propagation methodologies are defined at the application level and apply to all TP process rules. When Propagation is Run in a single process, together with Transfer Pricing rate calculations, the TP Engine first propagates rates and then calculates rates for the remaining un-priced records.

> **Note**
>
> Use the skip non-zero rates selection option in the TP Process to avoid overwriting TP rates on records that have either propagated rates or rates previously populated.

The Propagation process refers to details on the individual instrument records to determine if Transfer Rates are eligible for propagation. For example,

- If an instrument is a fixed rate, then propagate.
- If an instrument is adjustable, and a repricing has not occurred between the prior period and current period, then propagate.

In addition to the general logic, if Transfer Rate calculation is also selected in the same TP Process, then an additional test to confirm whether the TP Method for the applicable products is a “Bulk update” TP method then additionally, the propagation does not happen for related instrument records.

The test to determine if a TP method is a bulk method is as follows:

Spread from Interest Rate Code with mid-period repricing not selected.

Moving Average

Spread from Interest Rate Code or Redemption Curve with mid-period repricing not selected and Origination Date = As-of-Date.

The Standard Transfer Pricing Process allows you to propagate Transfer Rates and Adjustment Rates, and the Stochastic Transfer Pricing Process allows you to propagate Options Costs and Adjustment Rates. Depending upon your requirements, you can choose to propagate prior period results on the Transfer Pricing Calculation Selection block.

The main goal of using propagation is to improve 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.

> **Note**
>
> The Transfer Pricing engine propagate rates for instruments where it finds a matching ID_NUMBER for the As_of_Date data set and the selected prior period data set. The engine supports maximum ID_NUMBER up to 19 digits.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.6.2 Transfer Pricing Process and Audit Options

Transfer Pricing Processes provide you with three audit options: Detailed Cash Flow, Forward Rates, and 1 Month Rates. While the Detailed Cash Flow audit option applies to both Standard and Stochastic TP Processes, the Forward Rates and 1 Month Rates audit options are applicable only to Stochastic TP processing. From Release 8.1.1, FTP Audit will also capture the prepayment method used, IRC, Model along with other details like TP method, adjustment method, and so on.

By selecting the Detail Cash Flow option in the Transfer Pricing Process Audit block, you can audit daily cash flow results generated by the Oracle Funds Transfer Pricing application. Selecting this option writes out all cash flow and repricing events that occur for processed records. The user, based on the value input for the Number of Records, determines the number of records written.

The relevant financial elements for each instrument record and the cash flow results are stored in the FSI_O_PROCESS_CASH_FLOWS table.

> **Note**
>
> Propagation plus Detail Cash Flow output should not be checked in the same process, as this may result in less than the expected number of instrument records being output.

After processing cash flows from Transfer Pricing, follow these steps to view the audit results:

- Determine the value of the ID Numbers for the instruments that you want to process.
- View data by querying the FSI_O_PROCESS_CASH_FLOWS table with Oracle SQL Developer, SQL Plus, Toad, or an equivalent SQL query tool.

11.6.3 SQL Query Results

The results of running a SQL query can be exported to an Excel file for validation.

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:

Table 11-14 Financial Elements Written During Audit

Financial Element	Description
Initial Event	
100	Ending balance. The final balance on the payment date, after the payment has occurred.
430	Interest cash flow.
210	Total principal runoff, including scheduled payments, prepayments, and balloon payments.
60	Beginning balance. Starting balance and payment date, before payment.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-14 (Cont.) Financial Elements Written During Audit

Financial Element	Description
120	Runoff Net Rate. The rate at the time of payment, weighted by ending balance. To view the actual rate, divide financial element 120 by financial element 100.
490	A discount factor used to determine the present value of cash flow on the payment date.
Initial Event	
250	Par balance at the time of repricing.
280	Before Reprice Net Rate. Rate before repricing, weighted by repricing balance. To determine the 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 repricing balance. To determine the true rate, divide this financial element value by financial element 250.
Initial Event	
100	The initial balance at the start of processing.
120	The initial net rate at the start of processing.

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 table describes the different cash flow codes:

Table 11-15 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 the tease period).
8	Reprice during the tease period.
22	Reprice and payment events together (not during the tease period).
10	Reprice and payment during the tease period.

11.6.4 Data Verification

You can export results from your SQL query into Excel format and use Excel to validate the results. If the cash flows do not behave as expected, examine instrument table data or your assumptions.

For more information on Cash Flow Calculations, see the OFS Cash Flow Engine Reference Guide.

11.6.5 Transfer Pricing Process Rule and Calculation Modes

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

Copyright © 2011, 2026, Oracle and/or its affiliates.

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 calendar period end 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. After 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:

Embedded Rate Risk Spread = Total Rate Risk Spread - Current Rate Risk Spread

> **Note**
>
> 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 nor take drastic actions in response to profit or losses caused by the embedded rate risk.

11.6.6 Transfer Pricing Process and Migration Options

The purpose of the Ledger Migration process is to generate dollar charges or credits for funds provided or used for a combination of dimensions. The information necessary to generate these charges or credits (through transfer rates, adjustments, and option cost processing) originates from the Instrument Tables and the results are inserted into the Management Ledger table and are available for use in the calculation of profitability and risk measures. Within the Transfer Pricing Process definition screen, on the "Migration" block, users can select from two options for the charge/credit method:

- **Account Level**: The Account Level method sums the charge/credit amounts computed at the individual instrument level (based on the instrument's current or average book balances) and groups the results by the set of selected dimensions and migrate the amounts, together with the weighted average transfer rates to the Management Ledger table.
- **Ledger Level**: The Ledger Level method computes the weighted average transfer rates from the instrument data and will migrate these values to the Management Ledger table. The migration process then multiplies the weighted average transfer rates by the Ending or Average balances on the Management Ledger Table to arrive at the TP charge or credit amounts.

The following Migration – Charge/Credit Methods are available:

With both methods, the following rows are created for each product (and a combination of selected dimensions). An offset entry to the funding center (offset org unit) is also created.

When Transfer Rate is selected (based on Standard or Remaining Term option):

- Financial Element 170, Average Transfer Rate

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Financial Element 450, Charge/Credit
- Financial Element 172, Average Remaining Term Transfer Rate
- Financial Element 452, Charge/Credit Remaining Term

When Adjustments are selected (based on the population of noted adjustment type):

- Financial Element 174, Average Liquidity Adjustment Rate
- Financial Element 414, Liquidity Adjustment Charge/Credit
- Financial Element 175, Average Basis Risk Cost Rate
- Financial Element 415, Basis Risk Cost Charge/Credit
- Financial Element 176, Average Pricing Incentive Rate
- Financial Element 416, Pricing Incentive Charge/Credit
- Financial Element 177, Average Other Adjustment Rate
- Financial Element 417, Other Adjustment Charge/Credit

When Option Costs are selected: (based on Standard or Remaining Term option):

- Financial Element 171, Average Historic Option Cost
- Financial Element 451, Historic Option Cost Charge/Credit
- Financial Element 173, Average Current Option Cost
- Financial Element 453, Current Option Cost Charge/Credit

When All in TP Rate is selected (based on formula for all in TP rate):

- Financial Element 169, Average All in TP Rate
- Financial Element 421, All in TP Charge/Credit

Overview of the Ledger Migration Process:

- 169 - Average All in TP Rate
- 421 - All in TP Charge/Credit

Oracle Funds Transfer Pricing then writes the result as dollar charges/credits to the Management Ledger table using the following financial elements:

- 421 - All in TP Charge/Credit

> **Note**
>
> For a given combination of Organizational Unit and Product dimensions (or any other combination of dimensions), only one row should exist for the associated rate (170 - 177) and charge or credit amount (414-417 or 450-453). An offset posting to the Offset Org Unit or Funding Center, is also made for each posting.

Oracle Funds Transfer Pricing provides great flexibility in the Ledger Migration Process and the generation of corresponding charges and credits. Users can specify Ledger Migration for a combination of an extended list of dimensions, including organizational unit, product, channel, GL account, geography, or any other dimension that is part of the key dimension set.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> Only the key dimensions are available for inclusion during the migration process. This is because Oracle Funds Transfer Pricing displays only the processing key dimensions in the user interface.

You can choose to migrate the Transfer Rate, Adjustment Amounts, or the Option Costs, within the respective Standard or Stochastic Transfer Pricing Process.

11.6.7 Transfer Pricing Process and Calculating Economic Value

You can choose to calculate Economic Value as part of a Standard Transfer Pricing Process by selecting the Economic Value calculation element. This calculation option refers to the Economic Value assumptions defined within the Transfer Pricing Rule and is also eligible for Alternate Rate Output mapping. Additionally, there are seeded output columns available corresponding to each of the seeded interest type elements.

All Transfer Rate types (Transfer Rate, Transfer Rate Alt, Remaining Term Transfer Rate, and Remaining Term Transfer Rate Alt) will be written to a single/shared column, Economic Value Transfer Rate (EV_TP_RATE). If it is necessary to store more than one of these EV outputs, Alternate Rate Output Mapping can be used. Each type of Adjustment Rate is mapped to its corresponding EV column. For example:

- Economic Value Liquidity Premium Rate - EV_LIQ_PREM_RATE
- Economic Value Basis Risk Rate - EV_BASIS_RISK_RATE
- Economic Value Pricing Incentive Rate - EV_PRIC_INC_RATE
- Economic Value Other Adjustment Rate - EV_OTH_ADJ_RATE
- Economic Value Other Adjustment Alternate Output - EV_OTH_ADJ_RATE_ALT

Also, the All-in TP Rate is mapped to a corresponding EV column.

Economic Value All in Transfer Rate - EV_ALLIN_TP_RATE

The output format for the Economic Value calculation (inputs defined through the Transfer Pricing rule) is as follows:

- For Assets:
  - Economic Value = MV - BV
- For Liabilities:
  - Economic Value = BV– MV

  Where:
  - BV = Book Value = CUR_BOOK_BAL
  - MV = Market Value = Net Present Value of Principal (FE210) and Interest (FE430) Cash Flows

> **Note**
>
> For Adjustable Rate records, the calculation assumes maturity at the first reprice date. In this case, the Repricing Balance (FE250) is additionally used to derive the final principal cash flow amount.

Copyright © 2011, 2026, Oracle and/or its affiliates.

In addition to the calculation logic, users can specify the following two parameters:

- **Interest Only**: If this option is selected, the Net Present Value calculation considers only the Interest Cash Flows (FE430). In this case, the output format is as follows: Economic Value = MV.
- **Exclude Accrued Interest**: If this option is selected, the first interest cash flow will be computed from the As-of-Date to the Next Payment Date. The resulting market value will reflect the clean price. The system uses the accrual factor specified in the instrument data instead of defaulting to Actual/Actual (A/A).

For forward starting instruments, which are instruments that are not yet on the balance sheet, that is, ORIGINATION_DATE > AS_OF_DATE, the logic for computing Economic Value is as follows: Economic Value =MV

> **Note**
>
> For forward starting instruments where ORIGINATION_DATE > AS_OF_DATE, the initial Principal Cash Flow (FE210) will be negative, representing the cash outflow. For such instruments, the ORG_BOOK_BAL and CUR_BOOK_BAL should be the same since the instrument is coming into existence in the future.

11.6.8 Stochastic Transfer Pricing Process and Option Cost Parameters

In addition to transfer rates (created by a Standard TP Process), the Stochastic Transfer Pricing Process allows you to calculate the cost of options that are associated with your instruments. If you want to calculate option costs, you need to define the parameters used in option costing within the Stochastic Transfer Pricing Process.

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, the borrower can prepay many mortgages at any time without penalty. In effect, the lender has granted the borrower an option to buy back the mortgage at 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 (or floors) which limit its maximum (or 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 Funds Transfer Pricing uses the Monte Carlo technique to calculate the option cost. Oracle Funds Transfer Pricing calculates and outputs two spreads and the related option cost (rate) which is the difference between these two spreads. The spreads are as follows:

- Static spread (HISTORIC_STATIC_SPREAD)
- Option-adjusted spread (HISTORIC_OAS)

The option cost is derived as follows:

- Option cost (rate) = OAS – Static Spread

Copyright © 2011, 2026, Oracle and/or its affiliates.

The Option Cost calculation process will also compute the charge/credit amount using the Option Cost (rate), Average Balance, and Accrual Factor and will write the result to the following column:

- CHARGE_CREDIT_OCOST

The static spread is equal to the margin and the OAS to the risk-adjusted margin of an instrument. Therefore, the option cost quantifies the loss or gain due to risk. For more information on Transfer Pricing Option Cost, see the OFS Cash Flow Engine Reference Guide.

11.6.9 Stochastic Transfer Pricing Process and Stochastic Rate Indexing Rules

The Stochastic Rate Indexing rule is one of the assumption rules that you need to select within a Stochastic Transfer Pricing Process to calculate option costs.

The purpose of the Stochastic Rate Indexing Rule is to establish relationships between a risk-free Interest Rate Code (IRCs) and other interest rate codes or Indexes. The Stochastic Rate Indexing 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.

During Stochastic FTP processing, the system generates future interest rates for the valuation curve you selected, which are then used to derive the future interest rates for any Index associated with 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 the valuation of instruments associated with the derived IRCs or Indexes. As the risk-free rates change, the non-risk-free interest rates change accordingly.

11.7 Define and Execute a Break Identification Process

Break Identification processing should be run if automatic break detection is the preferred approach to populating the break events table (FSI_D_BREAK_FUNDING_CHARGES). Users also have the option to populate this table manually during data loading or use a hybrid approach. The Break Funding Charges table is the source table for calculating breakage charges.

To complete the breakage charge calculation process, after populating breakage data, the Break Funding Charges table should be selected in a Transfer Pricing Process as the source table, and the calculate Adjustments option should be selected under Calculation Selection. Finally, an appropriate Adjustment Rule (containing Breakage Charge assumptions), should be selected in the TP Process to apply the desired Breakage Charge calculation method.

The Break Identification Process supports the following Break Types:

- Full Break (type 1)
- Partial Break (type 2)
- Change in Attributes (type 3)
- Partial Break + Change in Attributes (type 5)

Assumptions:

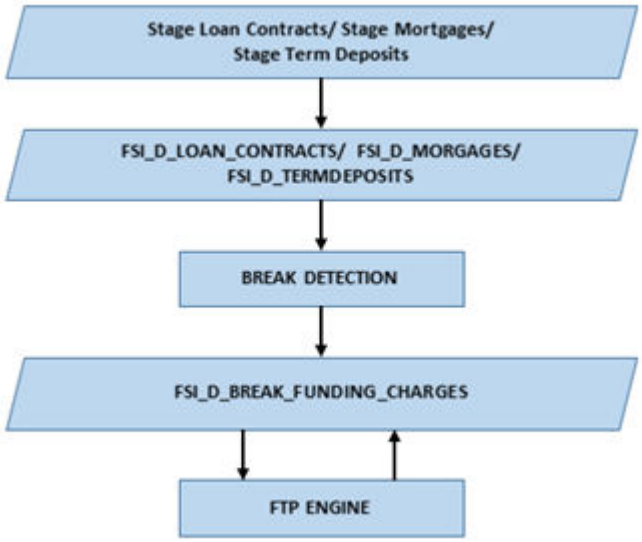
1. Instruments that are fully repaid or terminated will not be available in the instrument table for the current period. If users choose to include fully re-paid instruments, in the current data, these records should have a current balance of zero and account open flag = No.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Records will be identified and compared uniquely based on the ID Numbers. It is assumed that the ID Numbers will not be duplicated for a given period within an instrument table.
3. Both current and prior period data exist in the same instrument table. The current period refers to As-of-Date specified in FTP Application Preferences. A prior period is calculated based on assumptions given in the Break Identification Process.
4. When the Nearest Prior Date option is selected in the Break Identification Process for determining the prior period as of date, it is assumed that the process has been previously executed and an entry exists in the Process Run History table.

The following data flow illustrates how Break Identification processing works in the context of the overall staging and instrument table design:

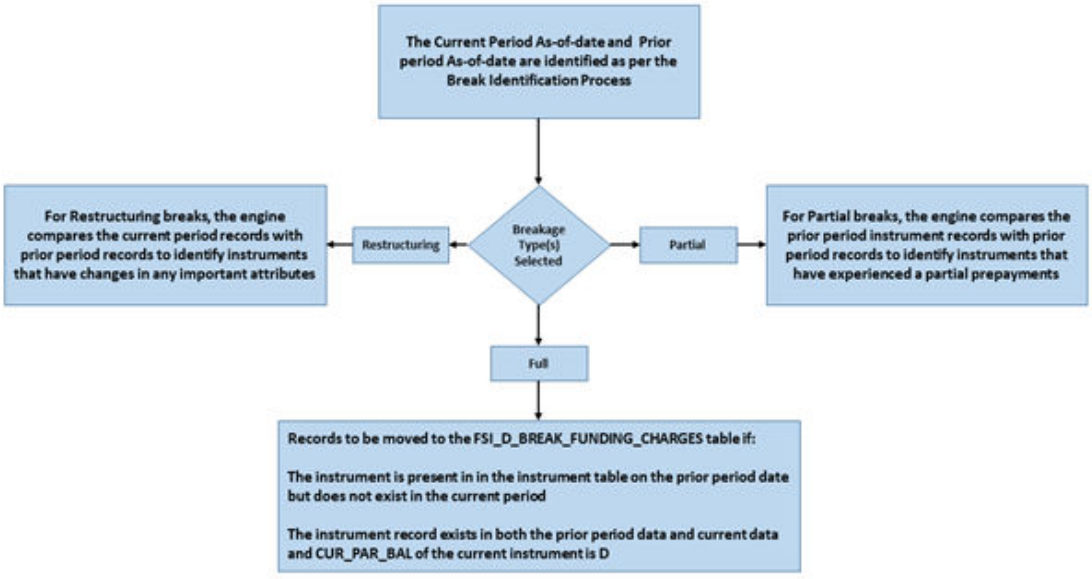
Figure 11-33 Break Identification Process Flow



The following figure shows the overview of the Break Identification process:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 11-34 Overview of Break Identification Process



- Full Breaks
  The following accounts are considered as full breaks, fully repaid, or terminated:
- Partial Breaks
  For partial breaks, the engine compares the prior period-instrument records with current period records to identify instruments that have experienced a partial prepayment.
- Change in Attributes Break
  For Change in Attributes Break, the engine compares the current period records with the prior period records to determine if any changes are made to critical attributes (other than balances). If any of the critical attributes have changed then it qualifies as a break event.

11.7.1 Full Breaks

The following accounts are considered as full breaks, fully repaid, or terminated:

- The Instrument is present in the instrument table on the prior period date but does not exist in the current period and the maturity date of the prior period record is greater than the current period as of the date.
- The Instrument record exists in both the prior period data and current data and the CUR_PAR_BAL of the current instrument is 0 and the account open flag is NO.

For the above scenarios, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:

- BREAKAGE_FLG = 2 (*Source record only) (External break will populate flag=1)
- BREAKAGE_TYPE_CD = 1
- BREAKAGE_AMOUNT = CUR_PAR_BAL (prior period)

Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:

- If Prior record NEXT_PAYMENT_DATE > Current AS_OF_DATE then:
  - CUR_PAR_BAL = Prior Period CUR_PAR_BAL

Copyright © 2011, 2026, Oracle and/or its affiliates.

  - CUR_BOOK_BAL = Prior Period CUR_BOOK_BAL
- If Prior record NEXT_PAYMENT_DATE <= Current AS_OF_DATE then:
  - IF AFTER_PAYMENT_BALANCE is Not Null
    CUR_PAR_BAL = AFTER_PAYMENT_BALANCE

    CUR_BOOK_BAL = AFTER_PAYMENT_BALANCE
- IF AFTER_PAYMENT_BALANCE is Null
  CUR_PAR_BAL = Prior Period CUR_PAR_BAL

  CUR_BOOK_BAL = Prior Period CUR_BOOK_BAL
- AS_OF_DATE = AS_OF_DATE defined in Application Preferences
- BREAKAGE_AMOUNT = Prior Period CUR_PAR_BAL
- BREAKAGE_TYPE_CD = 1
- All additional fields are carried forward from the prior period record.

For Full Breaks, the Next Payment Date gets rolled forward till the Maturity Date. Break record has AS_OF_DATE = NEXT_PAYMENT_DATE.

If NEXT_PAYMENT_DATE <= AS_OF_DATE, then Next Payment Date = Next Payment Date + Payment Frequency/Multiplier. This should not exceed the Maturity Date.

If NEXT_PAYMENT_DATE <= AS_OF_DATE for the current period, then Break Amount = AFTER_PAYMENT_BAL from the prior record. This is used when you do not want to pay a break charge against the current period's scheduled principal payment.

11.7.2 Partial Breaks

For partial breaks, the engine compares the prior period-instrument records with current period records to identify instruments that have experienced a partial prepayment.

The Break Identification Process uses Identity Code to determine which record is current and which is prior. It is assumed the Prior record Identity Code will be < Current Period.

The following accounts are considered as partial breaks:

Case 1

If the NEXT_PAYMENT_DATE on the prior period record is less than or equal to the AS_OF_DATE of the current period record, then:

- To detect a partial break, compare the AFTER_PAYMENT_BALANCE of the prior period record with the CUR_PAR_BAL of the current period record. If the difference is more than the MINIMUM_BREAK_AMOUNT then the instrument is classified as a Partial Break. For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:
  - BREAKAGE_FLG = 2 (*Source - Current record only) (External break will populate flag=1)
  - BREAKAGE_TYPE_CD = 2
  - BREAKAGE_AMOUNT = AFTER_PAYMENT_BALANCE of prior record – CUR_PAR_BAL of the current record

  Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The as of date for the prior record is changed to the current as of date.
- The BREAKAGE_AMOUNT of the prior record =0
- The BREAKAGE_AMOUNT of the current record = AFTER_PAYMENT_BALANCE of prior record – CUR_PAR_BAL of the current record
- BREAKAGE_TYPE_CD = 2
- Specific Fields updated on prior break funding record:
  - Next Payment Date: If Next Payment Date <= AS_OF_DATE then Next Payment Date +Payment Frequency/Multiplier
  - Last Payment Date: If Next Payment Date <= AS_OF_DATE then Next Payment Date
  - Next Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date + Reprice Frequency/Multiplier
  - Last Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date
  - Remaining Number of Payments: If Next Payment Date <= AS_OF_DATE then Remaining Number of Payments -1
- All additional fields are carried forward from the prior period record.

Case 2

If the NEXT_PAYMENT_DATE on the prior period record is greater than the As-of-Date of the current period, then:

- • To detect a partial break, compare the CUR_PAR_BAL of the prior period record with the CUR_PAR_BAL of the current record. If the difference is more than the Minimum Break Amount, then the instrument is classified as a partial break.
  For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:
  - BREAKAGE_FLG = 2 (*Source - Current Record only)
  - BREAKAGE_TYPE_CD = 2
    - If both Partial and Change in Attributes are detected, then BREAKAGE_TYPE_CD = 5
  - BREAKAGE_AMOUNT = CUR_PAR_BAL of the prior period record – CUR_PAR_BAL of the current record

  Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:
  - Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The as of date for the prior record is changed to the current As-of-Date.
  - The BREAKAGE_AMOUNT of the Prior Record =0
  - The BREAKAGE_AMOUNT of the Current Record = CUR_PAR_BAL of prior record – CUR_PAR_BAL of the Current Record
  - BREAKAGE_TYPE_CD = 2

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> If it is NULL, 0, or 1, the breakage charge calculation will treat as a full break and will not correctly calculate BREAK_FUNDING_AMT_CHG.

  - RECORD_IND = -1 for Prior Record and 1 for Current Record.
  - All additional fields are carried forward from the Prior Period Record.

> **Note**
>
> Partial Breaks are also detected when there is an increase in balance.

Example - Partial Break:

- Prior Record:
  - As-of-Date = 30-SEP-2012
  - Next_Reprice_Date = 01-OCT-2012
  - Last_Reprice_Date = 01-SEP-2012
  - REPRICE_FREQ = 1M
- Current Record:
  - As-of-Date = 01-OCT-2012
  - Next_Reprice_Date = 01-NOV-2012
  - Last_Reprice_Date = 01-OCT-2012
  - REPRICE_FREQ = 1M

The break occurs on 01-OCT-2012 and a break is detected on this date. This is also the Application Preferences As-of-Date given by the user.

Records moved to FSI_D_BREAK_FUNDING_CHARGES will be:

- Prior Record:
  - Since Next_Reprice_Date (1-Oct-2012) <= active As-of-Date (1-Oct-2012)
  - As_of_Date = 01-OCT-2012
  - Next_Reprice_Date = 01-NOV-2012
  - Last_Reprice_Date = 01-OCT-2012
- Current Record:
  - As_of_Date = 01-OCT-2012
  - Next_Reprice_Date = 01-NOV-2012
  - Last_Reprice_Date = 01-OCT-2012
  - The engine can only process the modified record in FSI_D_BREAK_FUNDING_CHARGES when Last_Reprice_Date <= As_of_Date < Next_Reprice_Date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.7.3 Change in Attributes Break

For Change in Attributes Break, the engine compares the current period records with the prior period records to determine if any changes are made to critical attributes (other than balances). If any of the critical attributes have changed then it qualifies as a break event.

The Break Identification Process uses Identity Code to determine which record is current and which is prior. It is assumed the Prior record Identity Code will be < Current Period.

The following key attributes are compared when analyzing data for Change in Attributes Break:

- ACCRUAL_BASIS_CD
- ADJUSTABLE_TYPE_CD
- AMRT_TYPE_CD
- COMPOUND_BASIS_CD
- CUR_NET_RATE (for fixed-rate instruments)
- CUR_PAYMENT (for fixed-rate instruments)
- MATURITY_DATE
- NEG_AMRT_AMT
- NEG_AMRT_EQ_DATE
- NEG_AMRT_EQ_FREQ
- NEG_AMRT_EQ_MULT
- NEG_AMRT_LIMIT
- PMT_FREQ
- PMT_FREQ_MULT
- REPRICE_FREQ
- REPRICE_FREQ_MULT
- RESIDUAL_AMOUNT

The list of seeded attributes can be viewed in the FSI_BRK_DETECTION_COLUMN_LIST table. Note that some attributes such as CUR_PAYMENT are not in the seeded list, but are referred to conditionally along with other columns.

If Prior Record Adjustable Type Cd = 0 and Current Record Adjustable Type Cd = 0 and Prior Cur Payment <> Current Cur Payment, then consider a break event.

If you want to consider any change in current payment as a break event, regardless of adjustable type code, then Cur Payment can be appended to the list of seeded values.

If any of these fields are identified as changed while comparing current and prior records, the records are flagged as a break.

For this scenario, the Break Identification Process populates the source and Break Funding Charges instrument tables with the following:

- BREAKAGE_FLG = 2 (Source – Current record only)
- BREAKAGE_TYPE_CD = 3
  - If both Partial and Change in Attributes are detected, then BREAKAGE_TYPE_CD = 5

Copyright © 2011, 2026, Oracle and/or its affiliates.

* BREAKAGE_AMOUNT = CUR_PAR_BAL

Additionally, the FSI_D_BREAK_FUNDING_CHARGES table is populated as follows:

- Both the prior period record and the current record are copied into the FSI_D_BREAK_FUNDING_CHARGES table. The As of Date for the prior record is changed to the current as of date.
  - BREAKAGE_TYPE_CD = 3

> **Note**
>
> If it is NULL, 0, or 1, the breakage charge calculation will treat as a full break and will not correctly calculate BREAK_FUNDING_AMT_CHG.

- RECORD_IND = -1 for Prior Record and 1 for Current Record
- All additional fields are carried forward from the Prior Period Record

If prior record's Next_Reprice_Date <= active As_of_Date

- Next Payment Date: If Next Payment Date <= AS_OF_DATE then Next Payment Date +Payment Frequency/Multiplier
- Last Payment Date: If Next Payment Date <= AS_OF_DATE then Next Payment Date
- Next Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date + Reprice Frequency/Multiplier
- Last Reprice Date: If Next Reprice Date <= AS_OF_DATE then Next Reprice Date
- Remaining Number of Payments: If Next Payment Date <= AS_OF_DATE then Remaining Number of Payments -1

> **Note**
>
> Active As-of-Date means the Application Preferences As-of-Date, which is also the current record,'s As-of-Date. The change applies only to the prior record. The current record will be corrected from the source system.
> If an account is repriced daily and Break Identification is done for one month period (difference in As-of-Dates), then though account's (prior period record) As-of-Date is updated to latest As-of-Date, but if the reprice dates are just rolled over by 1 Day as per reprice frequency, then the reprice dates go out of sync with As-of-Date. That is why for these accounts, the logic is updated as, if after rolling forward by one period, if the dates are not in sync with As-of-Dates, they will be rolled forward till As-of-Date, only for those adjustable accounts where reprice frequency or payment frequency is less than period over which break identification is performed. For longer reprice frequency accounts, there is no change in the logic.

11.8 Ledger Migration Process

This section discusses the process for generating charges or credits, for funds provided or used, and their migration to the Management Ledger table when using the Ledger Level-Charge or Credit migration option in the TP Process.

This section provides a detailed description of how the information required for generating these charges or credits originates through Transfer Rate, Adjustment Rate, and Option Cost

Copyright © 2011, 2026, Oracle and/or its affiliates.

Processing from the instrument tables and how the results are inserted into the Management Ledger table.

- Overview of the Ledger Migration Process
  Ledger migration is the process of generating aggregated charges (expenses) and credits (revenues) for funds provided or used for a combination of dimensions and their migration to the Ledger table.
- Understand the Ledger Migration
  To understand the process of creating Average Transfer Rate, Adjustment Rates, Option Cost, and Charge/Credit rows in the Management Ledger table (financial elements 170/172, 174-177, 171/173, and 450/451, 414-417, 452/453, respectively), you need to make the following assumptions:
- Requirements for Successful Ledger Migration
  Successful Ledger Migration of Transfer Pricing Results requires correct configuration of the following parameters:
- Instrument Tables Accumulation
  The first operation in the ledger migration process is to accumulate all individual detail rows from the Instrument tables into a single row for each unique combination of Org Unit and Product dimensions in the Virtual Memory Table (VMT).
- Management Ledger Table Processing
  The first step in the Ledger Migration Process with respect to the Management Ledger table is to clear all the information stored in the table with financial elements 170 and 450 (172 and 452 if remaining term pricing is being used) for the particular combination of dimensions being used in the process.
- Transfer Pricing Accounts with Ledger-Only Data Source
  At this stage, all rows in the Management Ledger table that relate (directly or indirectly) to rows in the Instrument tables are accumulated into the VMT. However, the accumulation process still needs to deal with account types that are transfer priced using Ledger as the data source (as specified in the Transfer Pricing Rule). In this example, Product 10 is a Direct Transfer Price product with a Management Ledger balance of 200.00.
- Transfer Pricing Un-priced Accounts
  Accounts using the Un-priced Account method are a special case of direct transfer pricing in the Management Ledger table. The Un-priced Account transfer pricing methodology uses the WATR from other accounts to derive a WATR for the Unpriced account. This is accomplished by averaging the WATR for the component accounts, weighted by their relative LS Balances.
- Ledger Migration of Option Costs
  Ledger migration of option costs is similar to that of the transfer rate. However, there are no steps for calculating option costs directly on the Management Ledger Table. This is because the calculation of option cost is a cash-flow based method that requires the Instrument table data. Normally, option cost is represented in the Instrument table record as the difference between two columns, HISTORIC_ STATIC_SPREAD and HISTORIC_OAS (Option Adjusted Spread) and is expressed as a rate, in percent.
- Usage of Intermediate Tables in the Engine for Management Ledger
  Ledger migration with Management Ledger tables uses intermediate tables. Intermediate tables can be Global Temporary tables or Normal tables. The creation of an intermediate table depends on the Application Preferences' Debug setting. If the Debug setting is - Do not output any message, then Global Temporary table is created. For any other debug setting, a normal table is created. Global Temporary table, which is created, gets dropped at the end of execution. Usage of a global temporary table will increase the performance of execution since data does not last after execution. Normal tables are not dropped at the end. To drop the tables, execute the purge script at regular intervals. All intermediate table

Copyright © 2011, 2026, Oracle and/or its affiliates.

names will start with zML_GTT_<process sys id>_<slno>. The naming convention is the same for both the global temporary table and the normal table. The <process sys id> is the process definition ID and <slno> is a number of the intermediate tables.

- Migration Options- Functional Currency and Entered and Functional Currency
  FTP provides the option to the user to perform Ledger Migration in the Entered/Transaction currency (the one in the ISO_CURRENCY_CD column of the Instrument table – the currency in which the transaction takes place) or in the Functional currency. These are supported in the Management Ledger table. The selections for 'Functional Currency' and 'Entered and Functional Currency' are present in the Migration block of the Standard and Stochastic Processes. To provide control over how FTP multi-currency postings happen, you can select 'Functional' or 'Entered and Functional'.
- FSI_DB_INFO Dependency for Multi-Currency
  There is no mandatory connection or reference to multi_currency_enabled_flg or currency_type_enabled_flg from the FSI_DB_INFO table. Column Currency_Type is not present in the Management Ledger structure.

11.8.1 Overview of the Ledger Migration Process

Ledger migration is the process of generating aggregated charges (expenses) and credits (revenues) for funds provided or used for a combination of dimensions and their migration to the Ledger table.

The information necessary to generate these charges and credits (through Transfer Rates, Adjustments, and Option Cost Processing) originates from the instrument tables and results are inserted into the Ledger table (FSI_D_MANAGEMENT_LEDGER) You can select only one Ledger Table per FTP process. The engine will work only on one Ledger Table and never on multiple within any single FTP Process. Multiple Instrument tables are allowed as usual. Transfer pricing charge and credit information provides the basis for measuring net interest income contribution for a group of products, organizational units, or a combination of other dimensions, and is available for use in further calculations of profitability, risk forecasting, and planning.

Oracle Funds Transfer Pricing provides great flexibility in the ledger migration process and in the generation of corresponding charges, credits, and option costs. Users can specify ledger migration for a combination of an extended list of dimensions. This feature provides flexibility to users who are also using Oracle Profitability Management for profitability reporting across organizational, product, channel, geography, or other user-defined dimensions.

Additionally, Oracle Funds Transfer Pricing provides multi-currency support that allows you to generate charges or credits for funds based on entered and functional currency. You can choose to migrate the Transfer Rate, Adjustment Rates, or Option Costs by selecting the appropriate options on the Calculation Elements block of your Transfer Pricing Process rules. See: Transfer Pricing Process.

The following financial elements are used to Ledger Migration:

- 100 – Ending Book Balance (Input FE)
- 140 – Average Book Balance (Input FE)
- 170 - Average Transfer Rate – Pertains to Standard Term
- 172 - Average Remaining Term Transfer Rate – Pertains to Remaining Term
- 174 - Average Liquidity Adjustment Rate
- 175 - Average Basis Risk Cost Rate
- 176 - Average Pricing Incentive Rate

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 177 - Average Other Adjustment Rate
- 171 - Average Historic Option Cost – Pertains to Standard Term
- 173 - Average Current Option Cost – Pertains to Remaining Term
- 414 - Liquidity Adjustment Charge/Credit
- 415 - Basis Risk Charge/Credit
- 416 - Pricing Incentive Charge/Credit
- 417 - Other Adjustment Charge/Credit
- 450 - Transfer Rate Charge/Credit – Pertains to Standard Term
- 451 - Historic Option Cost Charge/Credit – Pertains to Standard Term
- 452 - Transfer Rate Charge/Credit Remaining Term – Pertains to Remaining Term
- 453 - Current Option Cost Charge/Credit – Pertains to Remaining Term
- 419 - Rate Lock Option Cost
- 459 - Breakage Charge

> **Note**
>
> Input FE's can be Average Book Balance, Ending Book Balance or even Custom Balance FE's can be used if setup as new FE's and selected in Application Preferences.

11.8.2 Understand the Ledger Migration

To understand the process of creating Average Transfer Rate, Adjustment Rates, Option Cost, and Charge/Credit rows in the Management Ledger table (financial elements 170/172, 174-177, 171/173, and 450/451, 414-417, 452/453, respectively), you need to make the following assumptions:

- All rows in the relevant Instrument tables have already been transfer-priced and/or assigned an option cost.
- All rows contain a valid rate in one or more of the following columns:
  - TRANSFER_RATE
  - TRAN_RATE_REM_TERM
  - LIQUIDITY_PREMIUM_RATE
  - BASIS_RISK_RATE
  - PRICING_INCENTIVE_RATE
  - OTHER_ADJUSTMENTS_RATE
  - CUR_OAS
  - HISTORIC_OAS
- Average Balance or Ending Balance (financial element 140/100) information has been loaded into the Management Ledger table with a dimensionality that matches the instrument table data being migrated.

This document describes the mechanics, which occur just after the Instrument tables transfer pricing or option cost calculations are completed successfully and just before Transfer Rate,

Copyright © 2011, 2026, Oracle and/or its affiliates.

Adjustment Rate, or Option Cost Ledger migration starts. For example, the mechanics that occur just after Instrument tables are populated with valid Transfer Rates and just before the Weighted Average Transfer Rate (WATR) and the Charge/Credit rows in the Management Ledger table are updated.

The Ledger Migration of option costs works on the same lines as Transfer Rate and Adjustment Rate migration. However, there are certain differences.

- Transfer Rate, Adjustment Rate, and Option Cost Calculation
  The Oracle Funds Transfer Pricing Engine calculates and writes Balance-Weighted Average Rates to the Management Ledger table, using Current Book Balance, Average Book Balance, or a User-Defined Balance from each instrument record to perform the weighting process.
- Charge/Credit Generation
  In addition to the calculation of the Weighted Average Rate values at the combination of the Organizational Unit and the selected Product dimensions, charge/credit generation involves the following steps:
- Direct Transfer Pricing of Ledger Balances
  Oracle Funds Transfer Pricing allows users to calculate transfer rates for ledger average or ending balances that do not have corresponding Instrument table records using the following transfer pricing methodologies:
- Ledger Migration and the Management Ledger Table
- Data Signage
  The Management Ledger table supports variable data signs. You can load data into the Management Ledger table in the following three variations:
- Management Ledger Table Editing Standards
  You should be careful while editing the Management Ledger table directly. If you ever get unexpected results in the Management Ledger table after Ledger Migration, then review the data you have entered.
- Ledger Migration and the Virtual Memory Table
  To calculate transfer rates at the Product dimension member level in the Management Ledger table, all rows in the Instrument tables must be accumulated to arrive at the Weighted Average Transfer Rate (WATR) and Weighted Average Adjustment Rates (WAAR) for each member. All data used in the ledger migration process passes through a table, called the Virtual Memory table (VMT), and built in the memory.

11.8.2.1 Transfer Rate, Adjustment Rate, and Option Cost Calculation

The Oracle Funds Transfer Pricing Engine calculates and writes Balance-Weighted Average Rates to the Management Ledger table, using Current Book Balance, Average Book Balance, or a User-Defined Balance from each instrument record to perform the weighting process.

The financial elements that the engine uses to write the weighted rates to the Management Ledger are as follows:

- 170 Average Transfer Rate
- 172 Average Rem Term Transfer Rate
- 171 Historic Option Cost
- 173 Current Option Cost
- 174 Average Liquidity Adjustment Rate
- 176 Average Pricing Incentive Rate

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 177 Average Other Adjustment Rate

11.8.2.2 Charge/Credit Generation

In addition to the calculation of the Weighted Average Rate values at the combination of the Organizational Unit and the selected Product dimensions, charge/credit generation involves the following steps:

- Aggregation of the corresponding average or ending balance records from the Management Ledger table for each Org Unit/Product dimension combination.
- Multiplication of the average or ending balance from the Management Ledger by the weighted average rates.
- Application of an accrual factor to de-annualize the amount.

Oracle Funds Transfer Pricing then writes the result as dollar charges/credits to the Management Ledger table using the following financial elements:

- 414 Liquidity Adjustment Charge/Credit
- 415 Basis Risk Charge/Credit
- 416 Pricing Incentive Charge/Credit
- 417 Other Adjustment Charge/Credit
- 450 Transfer Rate Charge/Credit
- 451 Historic Option Cost Charge/Credit
- 452 Charge/Credit Remaining Term
- 453 Current Option Cost Charge/Credit

> **Note**
>
> Instrument Charge/Credit (TP, Adjustments and Option Cost)', when 'Calculation Mode' = 'Remaining Term' or 'Standard Term', the selected values for 'Accrual Type' (Daily or Monthly) has an impact on the calculation of the Accrual period for Charge/Credit calculation.

11.8.2.3 Direct Transfer Pricing of Ledger Balances

Oracle Funds Transfer Pricing allows users to calculate transfer rates for ledger average or ending balances that do not have corresponding Instrument table records using the following transfer pricing methodologies:

- Moving Averages
- Spread from Interest Rate Code
- Redemption Curve
- Caterpillar
- Weighted Average Perpetual
- Un-priced Account
- Tractor method

Copyright © 2011, 2026, Oracle and/or its affiliates.

Direct Transfer Pricing of Ledger Balances will always have to be done on only one Ledger Table, be it Ledger Stat or Management Ledger, whichever is selected in the Process. Users will do direct transfer pricing for ledger balances that do not have corresponding Instrument table records. The migration is done for those ledger balances that have corresponding instrument table records.

Oracle Transfer Pricing also generates records in the Management Ledger table, which are posted to the organizational unit (Org Unit), designated as the Transfer Pricing Offset Unit (a special Treasury Unit). During this process, an offset charge or credit amount is calculated for each normal charge/credit posted at the intersection of the Organization Unit and Product dimensions in the processes outlined above.

The sum of the Org Unit charges and credits at the Product dimension member level is multiplied by -1 and posted to the offset Org Unit designated in the Offset Org attribute for the Org Unit dimension. After this processing is complete, the total entity level charges and credits net to zero.

> **Note**
>
> If no Offset Org Unit is specified, the offset amount will be posted to a default Org Unit (-99100). Users should be aware that the TP Migration process does not read the Offset Org ID attribute defined against any other dimension (Product, GL Account, and Common).

Financial Elements related to Direct Transfer Pricing of Ledger Balances:

- 100 – Ending Book Balance (Inputs to the calculations)
- 140 – Average Book Balance (Inputs to the calculations)
- 170 - Average Transfer Rate (Only one out of 170 and 172 will get generated at a time, depending on Standard or Remaining Term)
- 172 - Average Remaining Term Transfer Rate
- 450 - Transfer Rate Charge/Credit (Only one out of 450 and 452 will get generated at a time, depending on Standard or Remaining Term)
- 452 - Charge/Credit Remaining Term

> **Note**
>
> Input FE's can be Average Book Balance, Ending Book Balance or even Custom Balance FE's can be used if setup as new FE's and selected in Application Preferences.

11.8.2.4 Ledger Migration and the Management Ledger Table

A thorough understanding of the Ledger Migration process requires familiarity with the Management Ledger table standards such as data signage, editing standards, and WATR and charge/credit rows.

11.8.2.5 Data Signage

The Management Ledger table supports variable data signs. You can load data into the Management Ledger table in the following three variations:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Absolute**: All account types are positive, and all contra accounts are negative.
- **GAAP or Standard**: The signs reflect standard accounting principles, with revenue, liability, and owners' equity as negative, and expenses and assets as positive.
- **Reverse of GAAP (Standard) or User Defined**: Revenue, liability, and owners' equity are positive, and expenses and assets are negative.

11.8.2.6 Management Ledger Table Editing Standards

You should be careful while editing the Management Ledger table directly. If you ever get unexpected results in the Management Ledger table after Ledger Migration, then review the data you have entered.

WATR, WAAR's and Charge/Credit Rows.

The Weighted Average Transfer Rate (WATR), Weighted Average Adjustment Rates (WAAR), and the resulting charge/credit for funds are represented in the Management Ledger table by financial elements listed above.

- **Financial Elements 170/172 (WATR)**: If you select the Remaining Term calculation mode while defining the Transfer Pricing Process, then the financial element generated is 172. Otherwise, it is 170. Only one 170/172 row should exist for a given combination of Organization Unit and Product dimensions.
- **Financial Elements 450/452 (Charges/Credits for Funds)**: If you select the Remaining Term calculation mode while defining the Transfer Pricing Process Rule, then the system generates financial element 452. If not, it would be Financial Element 450 for the Transfer Rate Charge/Credit amount. Only one financial element, 450 or 452, should exist for a given combination of Organization Unit and Product dimensions.
- **Adjustment Rate Financial Elements 174-177 / 414-417**: The Adjustment Rate outputs are not impacted by the Standard Term / Remaining Term selection. The results will be the same in either case.

11.8.2.7 Ledger Migration and the Virtual Memory Table

To calculate transfer rates at the Product dimension member level in the Management Ledger table, all rows in the Instrument tables must be accumulated to arrive at the Weighted Average Transfer Rate (WATR) and Weighted Average Adjustment Rates (WAAR) for each member. All data used in the ledger migration process passes through a table, called the Virtual Memory table (VMT), and built in the memory.

This table exists only during the ledger migration process and the information is never written to disk, and thus it cannot be examined for problem-solving purposes. Understanding the operation of the VMT, however, is crucial to understanding the Ledger Migration Process.

The VMT comprises the following three types of columns:

- Organization Unit and Product dimension columns, which uniquely identify each row.
- Balance and WATR/WAAR columns to hold data accumulated from the Instrument tables.
- Balance and WATR/WAAR columns to hold data accumulated from the Management Ledger table and Instrument table calculations.

11.8.3 Requirements for Successful Ledger Migration

Successful Ledger Migration of Transfer Pricing Results requires correct configuration of the following parameters:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Application Preferences
- Dimensions
- Entered and Local Currency
- Transfer Pricing Rule
- Adjustment Rule
- Product / Source Selection
- Migration and Product Dimension Set Up
- Offset Org Unit
- Transfer Pricing Process
- Calculation Mode
- Charge/Credit Accrual Factor

Together these parameters determine the way Transfer Rate, Adjustment Rate, and Option Cost calculations are carried out for every instrument record.

- Application Preferences
  You must configure the following application preference parameters:
- Dimensions
  To be eligible for inclusion in the Ledger Migration Process, a dimension must exist and be actively populated with dimension values in both the Instrument tables and in the Management Ledger table. Given below is a list of dimensions available for inclusion in the Ledger Migration Process:
- Entered and Local Currency
  Oracle Funds Transfer Pricing provides you with the option of performing Ledger Migration and writing charges and credits in the entered or local currency, designated in the ISO_CURRENCY_CD column, or in the functional currency.
- Source of Currency and Exchange Rate Information
  Oracle Funds Transfer Pricing sources currency and exchange rate information from Rate Management > Currency and Currency Rates screens. Ledger migration should only be performed for currencies that are activated or enabled.
- Transfer Pricing Rule
  The Transfer Pricing Rule is used to define the transfer pricing and option cost methodologies for each product dimension member. While defining transfer pricing methodologies, ensure that all required supporting data for the method exists.
- Adjustment Rule
- Product/Source Selection
  Calculating and migrating Transfer Rates, Adjustment Rates and Option Costs for an entire product portfolio can be a time-consuming process. Source table selection or data selection through the Product Hierarchy option together with user data filters, allow you to reduce the ledger migration time as follows:
- Ledger Migration and Product Dimension Set Up
- Offset Org Unit
  During Direct Transfer Pricing of Ledger Balances and Ledger Migration, FTP generates records in the Ledger table that are posted to the Organization Unit designated as the Transfer Pricing Offset Unit (as defined via attribute within each Org Unit dimension member). During this process, an offset charge or credit amount is calculated for each normal charge/credit posted at the intersection of the Organization Unit and Product (and any other dimensions selected for migration).

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Transfer Pricing Process Rule
  The Transfer Pricing Process acts as a container for all the Ledger Migration parameters and submits them to the Transfer Pricing Engine as a processing job. A Transfer Pricing Process rule contains the following Ledger Migration specifications.
- Calculation Mode
  The choice of calculation mode, on the Transfer Pricing Process – Calculation Selection block, not only affects the transfer rate and option cost calculation processes, but also the migration process. It determines the results that will be migrated to the Management Ledger table. If the calculation mode is set to Standard then the following results are used in migration:

11.8.3.1 Application Preferences

You must configure the following application preference parameters:

- **As-of-Date**: Must match the period for which you are trying to migrate Transfer Rates, Adjustment Rates, and Option Costs.
- **Ledger Migration**: Rate Weighting Element - Select the instrument table balance to use for weighting the rates during the migration process. Choose from Average Book Balance, Ending Book Balance, or Custom Balance.
- **TP Charge/Credit Balance**: select the Balance to use for calculating the Charge / Credit Amount. When using the “Ledger” based migration option, choose from Ending Book Balance or Average Book Balance. For calculating instrument-level charge/credit amounts, you may also choose the Custom Balance option.

11.8.3.2 Dimensions

To be eligible for inclusion in the Ledger Migration Process, a dimension must exist and be actively populated with dimension values in both the Instrument tables and in the Management Ledger table. Given below is a list of dimensions available for inclusion in the Ledger Migration Process:

- Mandatory Dimensions:
  - PRODUCT (the required product dimension is based on your Application Preference selection)
- Other Available Dimensions:
  - ORGANIZATION UNIT
  - COMMON COA
  - GL ACCOUNT

11.8.3.3 Entered and Local Currency

Oracle Funds Transfer Pricing provides you with the option of performing Ledger Migration and writing charges and credits in the entered or local currency, designated in the ISO_CURRENCY_CD column, or in the functional currency.

11.8.3.4 Source of Currency and Exchange Rate Information

Oracle Funds Transfer Pricing sources currency and exchange rate information from Rate Management > Currency and Currency Rates screens. Ledger migration should only be performed for currencies that are activated or enabled.

Copyright © 2011, 2026, Oracle and/or its affiliates.

If currency code values that have not been activated are discovered in the ledger migration process, an error message is written to the process errors log, and the ledger migration process skips records with those values.

Calculation of Functional Currency Values

To calculate and write charge/credit values expressed in functional currency to the Management Ledger table, a situation in multi-currency implementations:

1. Choose between entered or functional Ledger Migration while defining the Transfer Pricing Process.
2. Derive charge/credit amounts in the entered or local currency first, using Transfer Rate and balance information expressed in those currencies, and then convert the calculated charge/credit values for the “As-of-Date” to the functional currency.
3. Assume the last date associated with the “As-of-Date” as the basis for Ledger Migration, and use currency exchange rates corresponding to that date to perform conversions to functional currency for charges and credits written to the Management Ledger table.
4. Use the following algorithm for Exchange Rate Access:
   - If the exchange rate exists, use the rate for the last day of the period being processed.
   - If no exchange rate exists for the last day of the period being processed, use the latest exchange rate available in the rates table for the period being processed.
   - If no exchange rate exists for the period being processed, use an exchange rate value of 1.

11.8.3.5 Transfer Pricing Rule

The Transfer Pricing Rule is used to define the transfer pricing and option cost methodologies for each product dimension member. While defining transfer pricing methodologies, ensure that all required supporting data for the method exists.

For example, if the selected method is spread from the Interest Rate Code, ensure that the corresponding yield curve has been properly defined and has been populated with rates.

11.8.3.6 Adjustment Rule

The Adjustment Rule is used to define logic for applying TP Rate Adjustments or Add-on Rates for each appropriate product dimension member.

11.8.3.7 Product/Source Selection

Calculating and migrating Transfer Rates, Adjustment Rates and Option Costs for an entire product portfolio can be a time-consuming process. Source table selection or data selection through the Product Hierarchy option together with user data filters, allow you to reduce the ledger migration time as follows:

- **Data Filters**: Allow you to transfer price or migrate to a ledger a subset of your portfolio.
- **Source/Product Selection**: This feature gives you the option of selecting the Instrument tables or individual products for ledger migration during a particular Transfer Pricing Process Run.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.8.3.8 Ledger Migration and Product Dimension Set Up

All Product dimensions (Product, Common COA, GL Account) contain an attribute, < accrual basis>, that is used to designate the accrual factor for a particular product used in calculating the charge or credit for funds. This attribute should be defined for all products when the user wishes to base charge and credit calculations on product-specific accrual factors rather than a single process-specific accrual factor defined at the Transfer Pricing Process Rule level.

11.8.3.9 Offset Org Unit

During Direct Transfer Pricing of Ledger Balances and Ledger Migration, FTP generates records in the Ledger table that are posted to the Organization Unit designated as the Transfer Pricing Offset Unit (as defined via attribute within each Org Unit dimension member). During this process, an offset charge or credit amount is calculated for each normal charge/credit posted at the intersection of the Organization Unit and Product (and any other dimensions selected for migration).

If no Offset Org Unit is specified through the attributes within each Org Unit Dimension member, the offset amount will be posted to a default Org Unit (-99100). The TP Migration process does not read the Offset Org ID attribute defined against any other dimension (Product, GL Account, and Common COA).

11.8.3.10 Transfer Pricing Process Rule

The Transfer Pricing Process acts as a container for all the Ledger Migration parameters and submits them to the Transfer Pricing Engine as a processing job. A Transfer Pricing Process rule contains the following Ledger Migration specifications.

The dimensions that you want to include in the Ledger Migration Process are as follows:

- The tables that are to undergo transfer pricing, adjustment rate, or option cost calculations.
- Filters (optional) that are to be applied to the rows in each table.
- Transfer pricing, Adjustment Rule or Prepayment Assumptions to be used.
- Option Cost calculation parameters.
- Ledger Migration Method (Ledger level or Instrument level).
- Charge/Credit accrual basis to be used.

11.8.3.11 Calculation Mode

The choice of calculation mode, on the Transfer Pricing Process – Calculation Selection block, not only affects the transfer rate and option cost calculation processes, but also the migration process. It determines the results that will be migrated to the Management Ledger table. If the calculation mode is set to Standard then the following results are used in migration:

- Transfer_Rate
- Adjustment Rates
- Historic Option Cost (Historic_Static_Spread - Historic_OAS)

Consequently, the transfer pricing engine generates results for the following financial elements:

- 170 Average Transfer Rate
- 171 Historic Option Cost

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 174 Average Liquidity Adjustment Rate
- 175 Average Basis Risk Cost Rate
- 176 Average Pricing Incentive Rate
- 177 Average Other Adjustment Rate
- 414 Liquidity Adjustment Charge/Credit
- 415 Basis Risk Charge/Credit
- 416 Pricing Incentive Charge/Credit
- 417 Other Adjustment Charge/Credit
- 450 Transfer Rate Charge/Credit
- 451 Historic Option Cost Charge/Credit

If the calculation mode is set to the Remaining Term, then the migration process uses the following result columns:

- Tran_Rate_Rem_Term
- Current Option Cost (Cur_Static_Spread - CUR_OAS)

Consequently, the transfer pricing engine generates results for the following financial elements:

- 172 Average Rem Term Transfer Rate
- 173 Current Option Cost
- 452 Charge/Credit Rem Term
- 453 Current Option Cost Charge/Credit

> **Note**
>
> Adjustment Rates are not affected by the calculation mode selection. Adjustment rates will be migrated as noted above under either Mode selection.

Charge/Credit Accrual Factor: Select the Charge/Credit Accrual Factor on the Transfer Pricing Process Migration block or, define the Accrual Factor as an attribute for each Product dimension member. In case no selection is made, an Accrual Factor of 30/360 is applied. Example of Transfer Rate Ledger Migration:

Ledger migration requires you to select, among others, the following options while creating and executing the Transfer Pricing Process:

- Select both the Instrument tables and the Management Ledger table as the SOURCE tables to be processed.
- Select the transfer rate calculation (optional if previously executed), adjustment rate calculation (optional if previously executed), and the ledger migration processing options. Selecting the transfer rate and/or adjustment rate calculation options leads to the generation of transfer rates or adjustment rates for all records in the Instrument tables and for those records in the Management Ledger table for which you have defined a transfer rate with a “Ledger” source type. Selecting the ledger migration processing option instructs the application to gather balances, transfer rate, and adjustment rate information, generate credits and charges for funds and output the results to the Management Ledger table.

Oracle Funds Transfer Pricing allows you to include multiple dimensions in the Ledger Migration process. However, to keep this description simple, the following example assumes

Copyright © 2011, 2026, Oracle and/or its affiliates.

that only two dimensions, the Organization Unit dimension, and the Product dimension, are selected to generate results. The following table displays the Instrument table data for this example.

Table 11-16 Instrument Tables (for example, FSI_D_MORTGAGES)

ORG_UNIT_ID	PRODUCT_ID	CUR_BOOK_BAL	TRANSFER_RATE
1	3	100	4.00
1	4	125	4.50
1	5	200	3.00
1	3	200	3.00

Table 11-17 Management Ledger Table

ORG_UNIT_ID	PRODUCT_ID	FINANCIAL_ELEM_ID	MONTH_xx
1	3	140	250.00
1	4	140	200.00
1	5	140	100.00
1	10	140	200.00
1	100	140	990.00

As you compare the Instrument tables and the Management Ledger table data, notice the following:

- Product IDs 3, 4, and 5 match in both tables. These Product IDs represent the simplest case of ledger migration.
- Product ID 10 does not exist in the Instrument tables. This example assumes that it is a ledger-only account that is transfer priced directly using an acceptable Management Ledger Table data source-only method (part of the assumption definition in the Transfer Pricing Rule).
- Product ID 100 does not exist in the Instrument tables. This example assumes that it is a ledger-only account that will be transfer priced using the Un-priced Account Methodology, based on Product IDs 4, 5, and 10. (This transfer pricing method is defined in the Transfer Pricing Rule.)

The ledger migration process essentially comprises the following two broad phases:

- Instrument Tables Accumulation
- Management Ledger Table Processing

However, this example illustrates the operation of the Ledger Migration process in general and that of the virtual memory table (VMT) in particular demonstrates the following possible variations of the ledger migration process and special cases:

- Transfer Pricing Accounts with the Ledger-Only Data Source
- Transfer Pricing Un-priced Accounts
- Ledger Migration of Transfer Rates Under Remaining Term Calculation Mode

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.8.4 Instrument Tables Accumulation

The first operation in the ledger migration process is to accumulate all individual detail rows from the Instrument tables into a single row for each unique combination of Org Unit and Product dimensions in the Virtual Memory Table (VMT).

In this example, Bal_x_TfrRate for Product 3 is calculated as follows:

(100 * 4.00) + (200 * 3.00) = 1,000.00 = Bal_x_TfrRate

The following table represents the VMT after Account table accumulation has taken place.

Table 11-18 VMT Post Instrument Table Accumulation

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LSBal x TfrRate
1	3	300.00	1000.00	
1	4	125.00	562.50	
1	5	200.00	600.00	

11.8.5 Management Ledger Table Processing

The first step in the Ledger Migration Process with respect to the Management Ledger table is to clear all the information stored in the table with financial elements 170 and 450 (172 and 452 if remaining term pricing is being used) for the particular combination of dimensions being used in the process.

The next step is Management Ledger table accumulation: the Virtual Memory Table (VMT) is populated with the balance information stored in the Management Ledger Table. The following table represents the VMT after the Management Ledger Table accumulation has taken place. The updates are shown in bold.

Table 11-19 VMT Post Management Ledger Table Accumulation

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LS Bal	LSBal x TfrRate
1	3	300.00	1000.00	250	
1	4	125.00	562.50	200	
1	5	200.00	600.00	100	

Management Ledger table processing involves the calculation of the Weighted Average Transfer Rate (WATR). The WATR is calculated by prorating the WATR by the ratio between the Account tables and the Management Ledger table balances as follows:

(Bal x TfrRate / Bal) * LSBal = LSBal x TfrRate

For example, the WATR for Line Item 3 is calculated as follows:

(1,000.00 / 300.00) * 250.00 = 833.33

The following table represents the VMT after the WATR calculation has taken place.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-20 VMT Post WATR Calculation

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LS Bal	LSBal x TfrRate
1	3	300.00	1000.00	250.00	833.33
1	4	125.00	562.50	200.00	900.00
1	5	200.00	600.00	100.00	300.00

11.8.6 Transfer Pricing Accounts with Ledger-Only Data Source

At this stage, all rows in the Management Ledger table that relate (directly or indirectly) to rows in the Instrument tables are accumulated into the VMT. However, the accumulation process still needs to deal with account types that are transfer priced using Ledger as the data source (as specified in the Transfer Pricing Rule). In this example, Product 10 is a Direct Transfer Price product with a Management Ledger balance of 200.00.

The following table represents a VMT with a direct transfer price product.

Table 11-21 VMT with a Direct Transfer Price Product

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LS Bal	LSBal x TfrRate
1	3	300.00	1000.00	250.00	833.33
1	4	125.00	562.50	200.00	900.00
1	5	200.00	600.00	100.00	300.00
1	10			200.00	1000.00

11.8.7 Transfer Pricing Un-priced Accounts

Accounts using the Un-priced Account method are a special case of direct transfer pricing in the Management Ledger table. The Un-priced Account transfer pricing methodology uses the WATR from other accounts to derive a WATR for the Unpriced account. This is accomplished by averaging the WATR for the component accounts, weighted by their relative LS Balances.

In this example, Product 100 is an un-priced account that is transfer priced based on Products 4, 5, and 10. First, as shown in the following table, a new row is added to the VMT and populated with the balance stored in the Management Ledger table.

Table 11-22 VMT with a New Row Displaying Management Ledger Table Balance

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LS Bal	LSBal x TfrRate
1	3	300.00	1000.00	250.00	833.33
1	4	125.00	562.00	200.00	900.00
1	5	200.00	600.00	100.00	300.00
1	10			200.00	1000.00
1	100			990.00	

Then, the WATR for Product 100 is calculated by computing the weighted average of the WATRs of Products 4, 5, and 10. The WATR for Product 100 is calculated as follows:

Copyright © 2011, 2026, Oracle and/or its affiliates.

(900 + 300 + 1,000)/(200 + 100 + 200) = 4.4

The VMT is then updated with the standard form of WATR

(990.00 * 4.4) = 4,356.00 = LSBal_x_TfrRate

The following table represents the VMT after the un-priced account has been transfer priced.

Table 11-23 VMT displaying the WATR of Un-priced Account

ORG_UNIT_ID	PRODUCT_ID	Bal	Bal x TfrRate	LS Bal	LSBal x TfrRate
1	3	300	1000.00	250.00	833.33
1	4	125	562.50	200.00	900.00
1	5	200	600.00	100.00	300.00
1	10			200.00	1000.00
1	100			990.00	4356.00

Calculation of Overall WATR (Financial Element 170)

After all the Instrument tables and the Management Ledger table information has been accumulated in the VMT, the overall WATR can be calculated for each Org Unit/Product dimension combination and posted to the Management Ledger table. The WATR is simply the sum of all component WATRs (represented in the VMT as LSBal x TfrRate).

For example, WATR is calculated as follows:

833.33 + 900.00 + 300.00 + 1,000.00 + 4,356.00 = 7,089.33= WATR

Generation of Charge/Credit for Funds (Financial Element 450)

After the overall WATR is known, the charge/credit for funds in any period is given by the formula:

WATR * Balance * Accrual Factor = Charge/Credit for Funds

As Oracle Funds Transfer Pricing stores WATR as WATR * Balance, this reduces to:

WATR * Accrual Factor = Charge/Credit for Funds

For example, Charge/Credit for Funds is calculated as follows:

7,089.33 * (30/360) = 590.77 = Charge/Credit for Funds

Ledger Migration of Transfer Rates Under Remaining Term Calculation Mode

The ledger migration process is identical under the Remaining Term calculation mode except that Financial Elements 452 and 172 are substituted for 450 and 170 respectively.

Note that under the Remaining Term calculation mode, the transfer rate source in the Instrument tables is Tran_Rate_Rem_Term.

11.8.8 Ledger Migration of Option Costs

Ledger migration of option costs is similar to that of the transfer rate. However, there are no steps for calculating option costs directly on the Management Ledger Table. This is because the calculation of option cost is a cash-flow based method that requires the Instrument table data. Normally, option cost is represented in the Instrument table record as the difference

Copyright © 2011, 2026, Oracle and/or its affiliates.

between two columns, HISTORIC_ STATIC_SPREAD and HISTORIC_OAS (Option Adjusted Spread) and is expressed as a rate, in percent.

Option Cost = HISTORIC_STATIC_SPREAD - HISTORIC_OAS

If option cost ledger migration is specified in the Stochastic Transfer Pricing Process, option cost is accumulated in the Virtual Memory table (VMT) and written to the Management Ledger table as Financial Element 171, Average Historical Option Cost. The corresponding charge/credit for Funds is written as Financial Element 451, Historical Option Cost Charge/Credit.

Option Cost Ledger Migration under the Remaining Term Calculation Mode

The option cost ledger migration process is nearly identical under the Remaining Term calculation mode, except that Financial Elements 453 and 173 are substituted for 451 and 171, respectively.

Note that under the Remaining Term calculation mode, the option cost source in the Instrument tables is the difference between the Cur_Static_Spread and Cur_OAS columns.

11.8.9 Usage of Intermediate Tables in the Engine for Management Ledger

Ledger migration with Management Ledger tables uses intermediate tables. Intermediate tables can be Global Temporary tables or Normal tables. The creation of an intermediate table depends on the Application Preferences' Debug setting. If the Debug setting is - Do not output any message, then Global Temporary table is created. For any other debug setting, a normal table is created. Global Temporary table, which is created, gets dropped at the end of execution. Usage of a global temporary table will increase the performance of execution since data does not last after execution. Normal tables are not dropped at the end. To drop the tables, execute the purge script at regular intervals. All intermediate table names will start with zML_GTT_<process sys id>_<slno>. The naming convention is the same for both the global temporary table and the normal table. The <process sys id> is the process definition ID and <slno> is a number of the intermediate tables.

For migration of data from instrument table to ledger tables, three intermediate tables are used. For example:

- zML_GTT_<process sys id>_1
- zML_GTT_ <process sys id>_2
- zML_GTT_<process sys id>_3

11.8.10 Migration Options- Functional Currency and Entered and Functional Currency

FTP provides the option to the user to perform Ledger Migration in the Entered/Transaction currency (the one in the ISO_CURRENCY_CD column of the Instrument table – the currency in which the transaction takes place) or in the Functional currency. These are supported in the Management Ledger table. The selections for 'Functional Currency' and 'Entered and Functional Currency' are present in the Migration block of the Standard and Stochastic Processes. To provide control over how FTP multi-currency postings happen, you can select 'Functional' or 'Entered and Functional'.

- If you choose Functional, this would assume that all Entered Balances in ML = Functional Balances. All non-functional currency balances from instrument data would be converted to functional currency and the same amount would be posted to both the Entered Balance and Functional Balance columns. The Functional currency is fetched from FSI_DB_INFO.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The currency conversion would be done based on the FSI_EXCHNG_RATE_DIRECT_ACCESS table.

- Similarly, if you choose Entered and Functional, then the assumption is that ML data contains detailed currency information and Entered Balance <> Functional Balance (except where entered and functional currency is the same). In this case, FTP would convert non-functional currency balances to functional currency for purposes of posting to the Functional Balance column, but would not convert the amount posted to the Entered Balance column and ISO_Currency_CD would equal the currency from the instrument data. This approach assumes that FE 100/140 data in ML is similarly loaded with a consistent approach. For example, if Functional and Entered is selected, the FE 100 or 140 data should also reflect multiple ISO_Currency_CD and Entered Bal <> Functional Bal

> **Note**
>
> Input Ledger Balance will always be available in Functional currency. Entered Balance can be <> Functional Balance, but Functional Balance should always be available for the Input FE's 100, 140, and Custom Input Balance FE's if any.

11.8.11 FSI_DB_INFO Dependency for Multi-Currency

There is no mandatory connection or reference to multi_currency_enabled_flg or currency_type_enabled_flg from the FSI_DB_INFO table. Column Currency_Type is not present in the Management Ledger structure.

Daily Charge/Credit - Migration

The Management Ledger supports daily charge credit postings. In Ledger Stat, each day of posting will be cumulative to the existing value, so over a month (MONTH_x columns) you will have 30 daily postings. Each day will be posted separately, so there will be full transparency around each posting that is made over a month. Three 'Accrual Type' drop-down lists in the Process screen are used specifically for each type of Charge/Credit.

Example:

FSI_D_MANAGEMENT_LEDGER

- If a ledger process is Run on 1-Jan-2000, results are posted for that with As of Date as 1-Jan-2000.
- If a ledger process is Run on 2-Jan-2000, results are posted in another row with As of Date as 2-Jan-2000.
- If a ledger process is Run on 15-Jan 2000, then results are posted in another row with As of Date as 15-Jan-2000.

11.9 Define Behavior Patterns

A prerequisite for transfer pricing your product portfolio is capturing instrument behavior. Payment and repricing characteristics for most instruments can be accommodated through the data attribute values in the Instrument tables. However, certain instruments may not have contractual cash flows or may have cash flows that are based on unique payment and repricing patterns that are too complex to be accommodated in the standard fields of the Instrument tables. Oracle Funds Transfer Pricing allows you to define custom behavior, payment, and repricing patterns to accurately model the unique characteristics for such instruments.

Copyright © 2011, 2026, Oracle and/or its affiliates.

In a user-defined Behavior pattern, you can assign a unique amortization type code to a set of payment tenors, and define Principal Cash Flow amounts (via percentages) for the following instrument types:

- Non-maturity Instruments
- Non Performing Instruments
- Devolvement and Recovery of Guarantees

In Funds Transfer Pricing, three TP Methods take Behavior Patterns (non-maturity type) as inputs:

- Tractor Method
- Caterpillar Method
- Weighted Average Perpetual Method

Each of these transfer pricing methods will generate Runoff amounts corresponding to the maturity profile defined in the behavior pattern. The Tractor Method requires replicating portfolio type non-maturity behavior patterns and the Caterpillar and Weighted Average Perpetual Methods utilize standard non-maturity behavior patterns.

- Behavior Pattern Structure
  There are three types of behavior patterns:
- Payment Events
  You must define one or more payment events to complete a behavior pattern. A payment event is a set of payment characteristics, which define the timeline and amount of a specific payment in the behavior pattern. While the payment types can change based on the pattern type selected, three inputs are required for all behavior patterns:

11.9.1 Behavior Pattern Structure

There are three types of behavior patterns:

- Non Maturity*
- Non-Performing
- Devolvement and Recovery

These behavior patterns differ in how they allow you to categorize cash flows based on the specific behavior type being modeled.

The following lists summarize the various payment types available for each type of behavior pattern:

- For Non Maturity behavior pattern, the payment types available are Volatile and Core.
- For Non Performing behavior pattern, the payment types available are Substandard, Doubtful, and Loss.
- Devolvement and Recovery behavior pattern, the payment types available are:
  - Sight Devolvement
  - Sight Recovery
  - Usance Devolvement
  - Usance Recovery

The non-maturity behavior pattern provides an additional option to create a Replicating Portfolio. The Tractor TP Method exclusively uses the replicating portfolio type pattern.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.9.2 Payment Events

You must define one or more payment events to complete a behavior pattern. A payment event is a set of payment characteristics, which define the timeline and amount of a specific payment in the behavior pattern. While the payment types can change based on the pattern type selected, three inputs are required for all behavior patterns:

- Term
- Amount
- Type

Principal Runoff Term

The Term of the principal runoff is determined by the input of a Tenor and Multiplier. For example, if you are defining the “Volatile” component of a non-maturity instrument, you would likely define the Runoff term for the first amount as “1 Day”. In this case, Tenor = “1” and Multiplier = “Day”

Amount

The cash flow amount is defined as a percentage of the end of the period balance. For example, in the case of Non-Maturity behavior pattern type, if the assumption for the Volatile Runoff amount is that 25% of the portfolio is to be categorized as "volatile", the input should be "25". The sum of the percentages for the Non-Maturity behavior pattern type must equal 100%. The "Total Percentage" is shown at the bottom of the Pattern detail area indicating the cumulative percentage.

Payment Type

The payment type selection categorizes all payment runoff into meaningful groupings. The principal Runoff type is carried forward by the Cash Flow Engine and can be identified in the cash flow results by the following financial element codes.

- 186 Writeoff Positive
- 187 Timing of Writeoff Positive
- 188 Writeoff Negative
- 189 Timing of Writeoff Negative
- 200 Non Maturity - Core Runoff
- 201 Timing of Non Maturity - Core Runoff
- 202 Non Maturity - Volatile Runoff
- 203 Timing of Non Maturity - Volatile Runoff
- 204 Devolvement Runoff
- 205 Timing of Devolvement Runoff
- 206 Recovery Runoff
- 207 Timing of Recovery Runoff
- 208 Non-Performing Asset Runoff
- 209 Timing of Non-Performing Asset Runoff

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.10 Define Payment Patterns

A prerequisite for transfer pricing your product portfolio is capturing instrument behavior. Payment and repricing characteristics for most instruments can be accommodated through the data attribute values in the Instrument tables. However, certain instruments may not have contractual cash flows or may have cash flows that are based on unique payment and repricing patterns that are too complex to be accommodated in the standard fields of the Instrument tables. Oracle Funds Transfer Pricing allows you to define custom behavior, payment, and repricing patterns to accurately model the unique characteristics for such instruments.

In a user-defined payment pattern, you can assign a unique Amortization type 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

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

- Payment Pattern Structure
  Oracle Funds Transfer Pricing allows you to build of the following payment patterns:
- Payment Events
  You must define one or more payment event to complete a payment pattern. A payment event is a set of payment characteristics, which define the timeline and amount of a specific payment in the payment pattern.
- 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.
- Absolute Payment Patterns
  Absolute Payment Patterns are commonly used for instruments that pay on a seasonal schedule, such as agricultural or construction loans that require special payment handling based on months or seasons.
- 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.
- Split Payment Patterns

11.10.1 Payment Pattern Structure

Oracle Funds Transfer Pricing allows you to build of the following payment patterns:

- Absolute
- Relative
- Split

These payment patterns differ in terms of how they address payment schedules, which determine whether calendar dates or periods determine the payment events constituting the pattern. Absolute patterns are defined with sets of payment characteristics scheduled on specific calendar dates. Relative patterns are defined with sets of payment characteristics scheduled periods.

Copyright © 2011, 2026, Oracle and/or its affiliates.

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

Additionally, 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 Funds Transfer Pricing's Payment Pattern interface supports simultaneous multiple-user access.

11.10.2 Payment Events

You must define one or more payment event to complete a payment pattern. A payment event is a set of payment characteristics, which define the timeline 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, two characteristics are required for all amortizing patterns.

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.

Table 11-24 Different 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 detailed instrument data. This percentage is defined by 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-24 (Cont.) Different Payment Methods

Method	Description
Absolute Payment	This is an input payment amount. This amount represents both Principal and Interest for a conventional payment type and represents only the principal for a level principal payment type. For both types of patterns, absolute value payment amounts are entered as gross of participation.
Interest Only	This is a calculated payment amount. An interest-only payment is calculated during processing as balance times rate times accrual factor.

11.10.3 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 interest-only for this type of payment pattern.

11.10.4 Absolute Payment Patterns

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

For example, a loan 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.

> **Note**
>
> You can define absolute Payment Patterns only up to a year. This is because all entries are automatically ordered by date and are scheduled in a single year rotation.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.10.5 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.

For example, in a four-year loan 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.

11.10.6 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 with two concurrent Amortization schedules. Each separate Amortization schedule is termed a timeline and assigned a percentage of the balance. A Split Pattern can constitute both Absolute and/or Relative Payment Patterns within itself.

11.11 Define Repricing Patterns

User-Defined Repricing Patterns provide a mechanism to capture the repricing structure of instruments whose rates change according to complex schedules which cannot be captured in the standard fields of account tables.

The User-Defined Repricing Pattern allows you to define multiple changes to various elements affecting repricing including:

- Rates
- Margins
- Frequency

A Repricing Pattern has the following major components:

- User-Defined Repricing Pattern
- User-Defined Repricing Event

> **Note**
>
> Repricing Patterns are a common capability within OFSAA applications, but FTP users should be aware that while the FTP application recognizes instruments with Repricing Patterns (custom ADJUSTABLE_TYPE_CD), the TP Engine does not behave differently when compared to standard adjustable-rate instruments. The behavior is the same because FTP always considers the information provided on the instrument record first. When processing adjustable-rate instruments, including standard and repricing pattern-based products, the FTP engine will refer to the LAST_REPRICE_DATE and NEXT_REPRICE_DATE to determine the relevant fixed-rate period. For pattern-based records, users should ensure that these dates are consistent with the Repricing Pattern Dates.
> This note applies to standard FTP processing. However, when calculating Option Costs using the Monte Carlo Engine in FTP, the Repricing Pattern information is used to determine the future rates, rate cap/floor information, and any associated pricing margins that needs to be applied over the life of the instrument record.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- 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.
- 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.
- Absolute Repricing Pattern
  The Absolute Repricing Pattern is used for instruments that are date dependent. Each specific date is a separate event.
- Relative Repricing Pattern
  The Relative Repricing Pattern is a series of Repricing Events that are driven by user-defined timelines. It is used for instruments where the repricing is determined by the elapsed time since origination. You specify the duration of each Repricing Period (frequency) and the number of times the event should occur (repeat) before calculating the next event in the pattern.
- Pricing with Balance Tiers
  Repricing Patterns include the option to set rates based on the balance of the instrument being priced. This selection is optional and users can select “None”, meaning there are no balance tiers needed, or they can select “Current Balance”. If Current Balance is selected, then users can define balance tier ranges and include different pricing details for each balance tier. This option is useful for applying pricing to products such as Savings Accounts or Money Market Accounts that pay different levels of interest-based on the account balance.

11.11.1 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. For more information, see the following sections:

- Defining Absolute Repricing Patterns
- Defining Relative Repricing Patterns

11.11.2 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 Funds Transfer Pricing. Only Oracle ALM, another Oracle Financial Services analytical application when assigning a rate at the origination of new business and transaction strategy records, uses this feature.

Copyright © 2011, 2026, Oracle and/or its affiliates.

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 is 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.

> **Note**
>
> You can change the Repricing Pattern type at any time during the creation 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

Define each event with a Repricing Type of either Flat Rate or Indexed Rate. The Repricing Types determine the event detail characteristic that is 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 list describes the event detail characteristics that are available when the Flat Rate Repricing Type is selected:

- **Net Rate**: The new Net Rate value.
- **Gross Rate**: The new Gross Rate value.
- **Transfer Rate**: The new Transfer Rate value.

The 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 list describes the event detail characteristics that are available when the Indexed Rate Repricing type is selected:

- **Interest Rate Code**: The reference Interest Rate is used as the Index Rate to set gross and net rates. This list of values is pulled from the current Historical Rates database.
- **Transfer Interest Rate Code**: Interest Rate used to calculate the Transfer Rate.
  - **Yield Curve Term**: Term used in Interest Rate Code Lookups; if left blank, defaults to the term until the Next Repricing; set with value and multiplier.
  - **Net Margin**: Added to the Index Rate to get the Net Rate.
  - **Gross Margin**: Added to the Index Rate to get the Gross Rate.
  - **Transfer Margin**: Added to the Index Rate to get Transfer Rate.
  - **Rate Cap Life**: The upper limit for Gross Rate.
  - **Rate Floor Life**: The lower limit for Gross Rate.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **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.

11.11.3 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. For more information, see Defining Absolute Repricing Patterns.

11.11.4 Relative Repricing Pattern

The Relative Repricing Pattern is a series of Repricing Events that are driven by user-defined timelines. It is used for instruments where the repricing is determined by the elapsed time since origination. You specify the duration of each Repricing Period (frequency) and the number of times the 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 repeat value of 3. This translates into an event that reprices every month for a duration of three 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 repeat value should be set to 999. This prevents wrapping, or the recycling of all the defined events until the life of the instrument runs out.

11.11.5 Pricing with Balance Tiers

Repricing Patterns include the option to set rates based on the balance of the instrument being priced. This selection is optional and users can select “None”, meaning there are no balance tiers needed, or they can select “Current Balance”. If Current Balance is selected, then users can define balance tier ranges and include different pricing details for each balance tier. This option is useful for applying pricing to products such as Savings Accounts or Money Market Accounts that pay different levels of interest-based on the account balance.

Balance tiered pricing can be applied to detailed instrument records or to aggregate balances. When applying to aggregate balances, it is important to define the mix percentage of the portfolio for which the pricing is applied. This is necessary because the original mix of accounts containing the related balance information is lost when the data is aggregated. The mixed percentage input solves this problem. When balance tiered pricing is applied to detailed accounts, the mix percentage input is not required.

Example:

Assume that the repricing pattern is defined with the following balance tiers:

Balance > 0 and < 25,000 then current rate = 0.00%, mix % = 50%

Balance >=25,000 then current rate = 2.5%, mix % = 50%

Scenario 1: The instrument table contains detailed account records

Account #1 = $10,000

Copyright © 2011, 2026, Oracle and/or its affiliates.

Account #2 = $10,000

Account #3 = $10,000

Account #4 = $10,000

Account #5 = $10,000

Account #6 = $50,000

The resulting rate will be [($50,000 x 0.00%) + ($50,000 x 2.50%)] / $100,000 = 1.25%

Scenario 2: The instrument table contains 1 aggregated instrument record

Account #1 = $100,000

> **Note**
>
> In scenario 2, the mix percentage is needed.

The resulting rate will be (0.00 x .50) + (2.50 x .50) = 1.25%

11.12 Perform Cash Flow Edits

The data in the Instrument tables must be clean, accurate, and complete before it is used to generate cash flows and for further processing. Oracle Funds Transfer Pricing provides seeded Cash Flow Edit Rules to edit (clean and prepare) Instrument table data. You can create multiple Cash Flow Edit Processes depending on the data to be cleansed. You can also view the actual results of Cash Flow Edits by accessing the audit data written into the FSI_PROCESS_ERRORS table.

You can also select the preview mode option so that you can preview the changes that will be made to the Instrument table data as a result of Cash Flow Edits before those changes are applied in the Instrument tables.

It is highly recommended that you create and run Cash Flow Edits before processing data to generate any type of cash flow-related results.

11.13 Create Interest Rate Codes

Oracle Funds Transfer Pricing uses historical interest rate information to transfer price your balance sheet. The final transfer rate or add-on rate assigned to the records in your instrument 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 Funds Transfer Pricing implementation.

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 Funds Transfer pricing, this is done using reference interest rates, called Interest Rate Codes.

Creating Interest Rate Codes is a mandatory step in the Oracle Funds Transfer Pricing process. Interest Rate Codes (IRC) are defined and managed within the Rate Management area of the application.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Oracle Funds 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, OFSAA Rate Management allows you to manage the term structure modeling parameters, such as volatility and mean reversion speed, as well as currency rates and economic indicators.

For more information, see OFSAA Rate Management.

- Interest Rate Codes and Rate Lookups
  A rate lookup is performed to derive a transfer rate for the appropriate date/term combination.

11.13.1 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 Funds 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 Funds 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 As-of-Date, the Last Reprice Date, the Origination Date or the TP Effective Date, depending on the method and the instrument characteristics.

If the yield curve term is expressed in months or years, the term will be converted internally during rate lookups 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, cubic spline, or quartic spline 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 and term combinations:

Table 11-25 Transfer Rates for Different Date and Term Combinations

Date	1 Day	1 Month	3 Months	1 Year
01/01/2010	2.00	3.00	4.00	5.00

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 11-25 (Cont.) Transfer Rates for Different Date and Term Combinations

Date	1 Day	1 Month	3 Months	1 Year
01/15/2010	2.10	3.10	4.10	5.10
01/31/2010	2.20	3.20	4.20	5.20
02/15/2010	2.30	3.30	4.30	5.30

The following table displays Date and Term Combinations for Lookup:

Table 11-26 Date and Term Combinations for Lookup

Date	Lookup Term	Yield Curve Date Used	Term Before	Term After	Rate	Comments
01/07/2010	60 days	01/01/2010	1 Month	3 Months	3.50	Rate is approximately halfway between 3 Months (91.26 Days) and 1 Month (30.42 Days).
11/30/2009	182 days	01/01/2010	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).
03/15/2010	2 Year	02/15/2010	1 Year	None	5.30	Uses the last point on Yield Curve.

11.14 Access 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 Audit Block, the detailed cash flow results are written to the FSI_O_PROCESS_CASH_FLOWS table.

Oracle Financial Services Funds Transfer Pricing also generates Audit Trail based on your selection. The TP process generates and stores the Audit Tail information for each instrument like TP Process Sys ID, TP Method Code, IRC Code, etc. This information is written to the FSI_O_FTP_AUDIT_TRAIL table. The purposes of this process are as follows:

- To provide metadata details for important calculated results.
- To minimize the impact on processing time when Audit is enabled.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- To make Audit output optional as this is dependent on your selection.

11.15 Access Transfer Pricing Interest Rate Audit Results

Forward Rates and 1 Month Rates audit results can be written to an audit table to facilitate validation of option cost results.

Within the Stochastic Transfer Pricing Process Audit Block, selecting the Forward Rates option allows you to audit the static spread calculations, and the 1 Month Rates option allows you to audit the option-adjusted spread calculations by writing out the different paths of one-month rates. Since 360 one-month rates are written out for each rate path, the process can be time-consuming.

The Forward Rates and 1 Month Rates audit results are written to the FSI_INTEREST_RATES_AUDIT table.

11.16 Analyze Results

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

Additionally, 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.

Table 11-27 Steps to Analyze Transfer Pricing Results

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

The weighting factor for CONVEXITY, DURATION, and MODIFIED_DURATION is Market Value. The weighting factor for YTM, CUR_WARM, and AVERAGE_LIFE are weighted by cur par bal.

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 either the Standard or Stochastic Transfer Pricing Process. This option should be selected together with a filter, which identifies the specific 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's impact and compare it to the results from prior periods.

Copyright © 2011, 2026, Oracle and/or its affiliates.

11.17 Review Processing Errors

There is always the possibility that errors may occur during the execution of a Transfer Pricing Process. A log of such errors is generated during processing and can be accessed by selecting the "View Log" link from both Standard or Stochastic Transfer Pricing Process summary page. Within this log, the report lists the specific transaction for which an error was generated and provides the internally generated identifier of the Transfer Pricing Process 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.

11.18 Reprocess Erroneous Accounts

While reviewing your results, you might discover accounts with invalid results that need to be reprocessed. Transfer Pricing Processes allow you to rerun a subset of information to make corrections to appropriate products when needed.

If you need to reprocess a portion of your instrument data, make sure that you reprocess all the Product dimensions members, across all instrument tables, particularly if you need to re-calculate the Ledger Charge / Credit amounts.

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

11.19 Reconcile the Data

Reconciliation is the process of comparing the information carried in the Instrument tables to the General Ledger balances (contained in the Management Ledger table).

The goal of the Transfer Pricing Process is to transfer price of your entire balance sheet, as represented on the General Ledger. Many ledger accounts have corresponding data in the Instrument 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 Instrument tables is to be compared to the General Ledger data carried in the Management Ledger table. That level can be one dimension (to reconcile for each general ledger account number, for example, General Ledger Account ID) or multiple dimensions (to reconcile for each General Ledger Account Number within each business unit, for example, General Ledger Account and Organization Unit).

The most common type of reconciliation is to compare the current balance of Instrument 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 Instrument 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.

Variances between the Instrument table and the Management Ledger table should be corrected. If the magnitude of the variances is high, plug entries should be created to force the reconciliation to zero.

Copyright © 2011, 2026, Oracle and/or its affiliates.

12

Transfer Pricing Concept

Over the past few decades, financial institutions, such as banks, have evolved as semi-autonomous lines of business. Consequently, management requires separate income statements and balance sheets for each line of business to assess its performance. However, creating separate income statements and balance sheets requires the division of the net interest income among the business units. Oracle Funds Transfer Pricing fulfills this need. Transfer Pricing is a mechanism for dividing the net interest income of a financial institution (such as a bank) among its constituent business units (such as the deposit, treasury, and credit groups).

Transfer Pricing makes use of transfer rates to divide the net interest income into manageable components by separately identifying the spread earned from interest rate risk and the spread earned from risks managed by the lines of business such as credit risk. The transfer rate for funds is an interest rate representing the value of those funds to a financial institution, that is, the interest rate at which the financial institution can buy or sell those funds in an open market.

The Transfer Rate provides a benchmark for determining whether the yield on a loan (an asset), is enough to cover the associated credit risk and operating cost, besides the cost of acquiring the funds. Additionally, a Transfer Rate for funds allows you to compare the total cost of each source of funds, such as deposits (a liability), to other funding opportunities, for example, money or capital market funds. In effect, you use a Transfer Rate to measure the profit contribution of an asset or liability.

The following table shows a bank balance sheet.

Table 12-1 Bank Balance Sheet

Asset	Liability
Less Transfer Cost of Funds/Spread on Assets	Less Cost of Funds/Spread on Liability
Less Operating Cost/Profit Contribution	Less Operating Cost/Profit Contribution

Most large banks have recognized the value of Transfer Pricing and it has been a part of their performance measurement systems for years. However, the gains from adopting a Transfer Pricing Framework depends on the maturity of the methodology used.

This chapter describes the traditional and matched rate approaches to funds Transfer Pricing. A description of how matched Rate Transfer Pricing overcomes the shortcomings of the traditional approaches is also provided. The chapter ends with a description of the role of matched Rate Transfer Pricing in the evaluation of interest rate risk.

Topics:

- Traditional Approaches to Transfer Pricing
- A New Approach to Transfer Pricing
- How Matched Rate Transfer Pricing Works?
- Evaluating Interest Rate Risk

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Traditional Approaches to Transfer Pricing
  Banks following the traditional approaches to Transfer Pricing applied a single transfer rate to the net volume of funds generated or consumed by a business unit. They are used either the average or the marginal cost of funds as the Single Transfer Rate.
- A New Approach to Transfer Pricing
  As the shortcomings of the traditional Transfer Pricing systems became obvious, the financial services industry began to search for alternatives. The best solution was developed and implemented by a few leading financial institutions in 1979 and 1980.
- How Matched Rate Transfer Pricing Works?
  The Treasury often administers the Matched Rate Transfer Pricing. The Treasury conceptually buys the funds from the deposit-gathering group and sells them to the credit group. Line officers get a rate quote representing either the cost of the funds they want to lend or the value of the deposits they are gathering. The spread between this quoted rate and the interest rate on the asset or liability is fixed at a known level and maintained for the life of the asset or liability. Any fluctuation in this spread, whether caused by changes in the asset or liability yield curves or in the funds transfer yield curve, is accumulated at the Treasury level.
- Evaluating Interest Rate Risk
  Matched rate transfer pricing divides the net interest income of your institution into three components: Lending, Deposit, and the Rate Risk Profit (or loss). The Rate Risk Profit is derived by subtracting all credits for funds (Funding Center Expense) from all charges for funds (Funding Center Income).

12.1 Traditional Approaches to Transfer Pricing

Banks following the traditional approaches to Transfer Pricing applied a single transfer rate to the net volume of funds generated or consumed by a business unit. They are used either the average or the marginal cost of funds as the Single Transfer Rate.

Until the 1960s, banks used the average cost of funds as the Single Transfer Rate, primarily for loan pricing. If the yield on a loan was higher than the average cost of funds, banks believed that the loan had a positive spread and made the loan. Over time, the problem with this approach became obvious. Regulated low rate deposits, such as Demand Deposit Accounts (DDA) and Savings Accounts, held the average cost of funds for many banks at a level well below the cost of the new funds. As a result, spreads on new volumes were nowhere near what had been expected. Moreover, the low average cost of funds tempted many banks to underprice loans, sometimes to the point where the true spreads on new volumes were negative.

Consequently, even a stable rate environment was potentially dangerous for banks using the average cost approach to transfer pricing because the balance sheet could grow while earnings dropped.

Recognizing that the use of an average cost of funds could result in unprofitable growth, most banks concluded they should use a Transfer Price reflecting their real cost of incremental funds. These banks used the cost of 30 or 90-day Certificates of Deposit (CDs) as the cost of marginal funds.

Pitfalls of the Traditional Approaches

The shortcomings of the Transfer Pricing approaches that advocate the use of a single transfer rate are:

- **Potential for Inadvertent Unprofitable Growth**: Banks assumed that using the marginal cost of funds would make it almost impossible to add volume at a negative spread. This is a fair assumption but it only applies to times when interest rates are stable.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Rate Risk Trap**: The single, marginal funds transfer rate led some financial institutions into a rate risk trap. In the 1960s and 1970s, because the yield curve was normal, long-term assets offered the largest spreads against a 30 or 90-day transfer rate. Therefore, some banks, and almost the entire savings and loan industry, borrowed short and lent long. Interest rates skyrocketed in 1979 and into the 1980s, and consequently, the margins disappeared.
- **Loss of the Credibility of Performance Measurement Systems**: Most banks were able to avoid the extreme interest rate risk exposure, which nearly destroyed the savings and loan industry. However, the use of a single marginal transfer rate undermined the credibility of performance measurement systems. Most line of business managers found that their bottom lines fluctuated wildly with interest rates. Since market interest rates were obviously beyond the control of line managers, they increasingly viewed profit goals for their units with skepticism.
- **Loss of Managerial Value**: The traditional approaches failed to offer any accepted (or politically acceptable) method for determining the net interest contribution of the different business units of a bank. Consequently, business unit profitability reporting lost its managerial decision-support value.

In summary, the traditional approaches to transfer pricing were acceptable when interest rates were stable. However, they lost most of their decision-supporting value after rates became volatile.

12.2 A New Approach to Transfer Pricing

As the shortcomings of the traditional Transfer Pricing systems became obvious, the financial services industry began to search for alternatives. The best solution was developed and implemented by a few leading financial institutions in 1979 and 1980.

This approach, called matched rate transfer pricing, uses multiple transfer rates. Assets and liabilities are given transfer rates that reflect their specific maturity and repricing characteristics. Matched rate transfer pricing resolves the problems inherent in traditional methodologies by:

- Clearly showing whether new volumes have a positive spread by using a marginal rate. This eliminates the potential for inadvertent unprofitable growth.
- Identifying potential rate risk traps in advance using a marginal rate. Additionally, the exposure of a bank to interest rate risk is identified and measured in a manner that makes it easier to manage.
- Ensuring that the performance measurement system is consistent, fair, and credible by using a Transfer Rate that reflects real funding opportunities currently available to the bank.

Matched rate transfer pricing achieves these objectives by dividing the interest rate spread into three components: credit spread, funding spread, and rate risk spread.

Example of Dividing Interest Rate Spread

Suppose a retail financial institution, a bank, for example, relies on a retail customer base for low-cost funds that have interest rates lower than funds purchased in money markets. It uses these funds to make loans that have a yield much higher than what the financial institution would pay for funds having the same maturity.

Consider a consumer loan that yields 200 basis points higher than what the financial institution would pay for funds having the same maturity. Suppose the bank decides to fund the loan with matched maturity funds, say, certificates of deposit that cost 100 basis points less than similar maturity funds purchased in money markets. Then, the bank will have a total interest rate spread of 300 basis points.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Matched rate transfer pricing divides this interest rate spread as follows. While the loan yields 200 basis points more than matched funding costs (transfer rate), the funds cost 100 basis points less than other alternatives (transfer rate). Therefore, the total spread of 300 basis points is the sum of a funding spread (Transfer Rate - The Cost of Funds) of 100 points and a credit spread (Yield on Loans - Transfer Rate) of 200 points.

However, if the financial institution funds the consumer loan with shorter-term deposits, then the spread would be larger than 300 basis points. The added spread result from taking interest rate risk (borrowing short and lending long) and is called Rate Risk Spread. The three components of the Interest Rate Spread can be seen by plotting the loan and deposit against the yield curve. However, the portion of the total spread derived from taking Interest Rate Risk can be volatile.

- Advantages of Matched Rate Transfer Pricing
  The main advantages of matched rate transfer pricing are as follows:

12.2.1 Advantages of Matched Rate Transfer Pricing

The main advantages of matched rate transfer pricing are as follows:

- **Stabilization of Business Unit Margins**: The use of multiple matched transfer rates stabilizes the margins of the different business units. Since assets and liabilities are either funded or sold to transfer pools with corresponding maturities or repricing periods, swings in interest rates do not affect the spread. Additionally, the division of the interest rate spread into credit, funding, and rate risk spreads ensures that the bottom line for a business unit reflects only that business and is within the control of the line management.
- **Decision Support**: Under the matched rate transfer pricing approach, the bottom line for a business unit is a fair basis for its performance measurement and management. For example, if some types of loans consistently fail to cover the cost of funds, operating costs, and credit risk, there is no reason to make those loans. It would be more profitable to buy bonds or to find other, more profitable lending opportunities. This is the reason why many banks no longer make small installment loans.
  Similarly, if the operating costs of gathering low-cost consumer deposits are too high, it may be more economical to purchase funds in money markets. This explains the growing number of branch closures, as well as the imposition of increasingly higher minimum balances on some types of consumer deposits.
- **Identifying Exposure to Interest Rate Risk**: Using matched rate transfer pricing, banks can identify their exposure to interest rate risk and its impact on their current earnings. Additionally, the banks can isolate the spread from rate risk exposure from their total spreads. This helps them clearly determine the profitability of their business units. Banks have found that the interest rate risk becomes increasingly manageable when isolated in a separate business unit. Under the matched rate transfer pricing approach, the rate risk exposure, and its impact on current earnings, is revealed in a new profit center called Treasury.

In summary, matched rate transfer pricing works well even when interest rates are volatile. It provides an approach to performance measurement that meets the decision making needs of both line managers (consistency, fairness, controllability) and executive managers (accuracy, flexibility). The financial services industry has recognized these benefits. Consequently, there is an increasing number of financial institutions that either have implemented or are in the process of implementing performance measurement systems based on matched rate transfer pricing.

Copyright © 2011, 2026, Oracle and/or its affiliates.

12.3 How Matched Rate Transfer Pricing Works?

The Treasury often administers the Matched Rate Transfer Pricing. The Treasury conceptually buys the funds from the deposit-gathering group and sells them to the credit group. Line officers get a rate quote representing either the cost of the funds they want to lend or the value of the deposits they are gathering. The spread between this quoted rate and the interest rate on the asset or liability is fixed at a known level and maintained for the life of the asset or liability. Any fluctuation in this spread, whether caused by changes in the asset or liability yield curves or in the funds transfer yield curve, is accumulated at the Treasury level.

The Treasury can manage the fluctuation in the spread in several ways, for example:

- Maintain a discretionary portfolio of assets and liabilities with the sole purpose of offsetting the risk that has been transferred from other business units.
- Use off-balance-sheet transactions, such as swaps and futures, to hedge risk.

Matched Rate Transfer Pricing requires more accounting discipline than Traditional Transfer Pricing approaches. However, it is a straightforward process and is applied in a logical manner, using standard principles of dual-entry accounting.

Matched Rate Transfer Pricing Example

Suppose a line officer wants to make a loan and is trying to decide on its pricing. The line officer is given a cost of funds that reflects the maturity and repricing characteristics of the loan. If it is to be a long-term, fixed-rate loan, the bank quotes the cost of the long-term funds that can be used to match that loan. Conversely, if the loan is to be short term, the line officer is quoted a short-term rate.

If the yield curve is normal, the transfer rate for a short-term loan is less than the rate for a long-term loan. The line officer then figures out how to price the loan to attain a target spread over the quoted cost of funds. When the loan is booked:

- The business unit of the line officer books a shadow liability equal in volume to the size of the loan, having a cost that equals the transfer rate that was quoted. This accounting transaction balances the books of the business unit and locks in a spread as long as the loan stays on the books.
- The books of the corporation must be balanced. Banks do this by creating a shadow asset with equal size and rate to the shadow liability. This shadow asset is housed in a separate business unit, usually Treasury.

The same type of accounting is applied to liabilities also. This type of accounting divides the bank's profits into three components: lending profit, deposit-gathering profit, and rate risk profit. These three components add up to the total profit of the bank.

To sum up, under the matched rate transfer pricing approach, banks attach a Matched Transfer Rate to an asset or liability when it is booked, using a standard, double-entry accounting approach. This Transfer Rate remains constant over the life of the asset or liability, stabilizing the spread for the line of business.

12.4 Evaluating Interest Rate Risk

Matched rate transfer pricing divides the net interest income of your institution into three components: Lending, Deposit, and the Rate Risk Profit (or loss). The Rate Risk Profit is derived by subtracting all credits for funds (Funding Center Expense) from all charges for funds (Funding Center Income).

Copyright © 2011, 2026, Oracle and/or its affiliates.

A net positive number implies that part of your interest margin is a result of any rate bets (or rate risk) your institution has taken. A negative number implies that you have incurred a loss due to rate risk.

Current and Embedded Rate Risks: The total rate risk profit (or loss) figure is made up from two sources:

- Current Rate Risk Profit: The result of the rate risk inherent in your current exposure. You can actively manage this profit through effective Asset/Liability management.
- Embedded Rate Risk Profit: The result of interest-rate bets. You can no longer manage this component of earnings because the relationships are contractual. All you can do is wait it out.

Embedded Rate Risks Example

Suppose a bank, on day one, raises $1,000 in the form of a one-year certificate of deposit at 4%. If the wholesale (open market) alternative to one-year funds costs 5% then, the matched transfer rate is 5%.

The bank then lends the $1,000 in the form of a five-year non-amortizing (bullet) loan at 10%. If the cost of five-year wholesale funds is 8% then the matched transfer rate for five-year funds is 5%.

This table shows the components of the bank's interest rate margin on day one:

Table 12-2 Bank’s Interest Rate Margin on Day One in case of Embedded Rate Risk

Income Statement Component	Rate	Transfer Rate	Spread
Asset	10.00%	8.00%	2.00%
Liability	4.00%	5.00%	1.00%
Funding Center Spread		3.00%	
Net Interest Margin	6.00%		

Over the next year, interest rates rise by 200 basis points. Now, the bank, eager to eliminate future rate risk, issues a new four-year $1,000 CD at 8.5%. However, the four-year transfer rate is now 9.5%.

This table describes the components of the interest margin for the bank after one year:

Table 12-3 Bank’s Interest Rate Margin after One Year in case of Embedded Rate Risk

Income Statement Component	Rate	Transfer Rate	Spread
Asset	10.00%	8.00%	2.00%
Liability	8.50%	9.50%	1.00%
Funding Center Spread		-1.50%	
Net Interest Margin	1.50%		

Although the bank is now perfectly matched from a current rate risk perspective (a four-year bullet loan funded by a four-year CD), it is losing 150 basis points at the funding center.

On day one, the bank took a rate bet by funding short. The bet was that one year from the loan origination date the bank would be able to raise four-year funds at less than the cost of funding

Copyright © 2011, 2026, Oracle and/or its affiliates.

the original five-year loan, or 8%. Since the four-year transfer rate on day one was 7%, when interest rates went up by 200 basis points, the bank was badly hit.

Although the net interest margin of the bank is still 150 basis points, the bank could have locked in a 300 basis point net interest margin for five years on day one if it had not taken a rate bet by issuing a five-year CD.

The loss of 150 basis points on the $1,000 loan is a result of the embedded rate risk taken by the bank. The bank can do nothing to eliminate embedded rate risk, except wait.

- **Measuring Current and Embedded Rate Risks**: Even though nothing can be done about Embedded Rate Risk, it is important to identify the impact of Embedded Rate Risk for planning ahead. For example, if a bank had a large profit in the funding center owing to Embedded Rate Risk, and was unaware of this, it can be lulled into a false sense of security. That bank might be surprised when this source of profit evaporates. Conversely, if a bank is experiencing a large loss in the funding center due to Embedded Rate Risk, and it can measure it, the bank might choose to wait it out rather than taking drastic and immediate actions.
- **Measuring Current Rate Risk**: You can measure Current Rate Risk by Transfer Pricing your entire balance sheet as if it were originated today. Everything should be Transfer Priced based on its remaining term. Under this method, a five-year CD with one year until maturity would receive the same Transfer Rate as a three-year CD with one year left.
- **Measuring Embedded Rate Risk**: The total rate of risk profit is made up of Embedded Rate Risk and Current Rate Risk. Embedded Rate Risk = Total Rate Risk Result - Current Rate Risk Result

Copyright © 2011, 2026, Oracle and/or its affiliates.

13

Common Rule Management Tasks

This chapter focuses on the Rule Management Tasks that are common across all rules in this application. The Rule Management Tasks that are common to business rules in this and other OFSAA applications.

Topics:

- The Rule Summary Page
- Search for Rules
- Create Rules
- View and Edit Rules
- Copy Rules
- Delete Rules
- Dependency Checking

You can perform these tasks from the Summary 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 Summary Page
  The Rule Summary page is the gateway to all rules and related functionality of the application. From there, you can navigate to other related pages.
- Search for Rules
  Search for a business Rule to perform any of the following tasks:
- Create Rules
  You create a Rule to specify the way you want a particular task or business process to be carried out by the application.
- View and Edit Rules
  You can view existing rules, and you can edit existing Rules, provided you have read and write privileges.
- Copy Rules
  You can copy rules to avoid having to enter data multiple times. This saves time and effort and also reduces mistakes.
- Delete Rules
  You can delete rules that are no longer needed. Once deleted, a rule cannot be retrieved.
- Dependency Checking
  You can check dependencies for rules to know where a particular Rule, Dimension, Member, Attribute, IRC has been used. In addition, this prevents accidental deletion of rules having dependencies.

Copyright © 2011, 2026, Oracle and/or its affiliates.

13.1 The Rule Summary Page

The Rule Summary 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 Summary page, you can perform simple queries on Folder, Rule Name, and in many cases, the dimension on which the rule is based.

Table 13-1 Page components for the Rule Summary Page

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Folder	Drop-Down List	Set in Application Preferences	Required - for filtering the rules under the folder	No – Only able to select from the 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.
Dimension	Drop-Down List	Set in Application Preferences		No - Only able to select from the presented list	N/A
Search	Button	N/A	N/A	No	Initiates rule search based on specified criteria.
Reset	Button	N/A	N/A	No	Restores default search criteria.
Add	Button	N/A	N/A	No	Initiates the Data or Ledger Loader rule creation process
(Rule) Name	Display Value	N/A	N/A	No	Mouse over shows the rule description and in some cases also displays the unique system id number.
Created By	Display Value	N/A	N/A	No	Who created the Rule version.
Creation Date	Display Value	N/A	N/A	No	When was the rule created?

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 13-1 (Cont.) Page components for the Rule Summary Page

Name	Type	Default Value	Required/ Optional	Updatable	LOV, additional information
Last Modified By	Display Value	N/A	N/A	No	Who last modified the rule?
Last Modified Date	Display Value	N/A	N/A	No	When the rule was last modified?
View	Icon	N/A	N/A	N/A	Opens the selected rule in read-only mode.
Edit	Icon	N/A	N/A	N/A	Opens the selected rule in edit mode.
Delete	Icon	N/A	N/A	N/A	Deletes the selected Rule.
Copy	Icon	N/A	N/A	N/A	Initiates the process of copying rules. Explained later in this document.
Run	Button	N/A	N/A	N/A	Initiates process for running Rules. Explained later in this document.
Pagination Options	Icon	Set in Global Preferences	N/A	N/A	Indicates the number of rows to display per page in the summary table.

13.2 Search for Rules

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

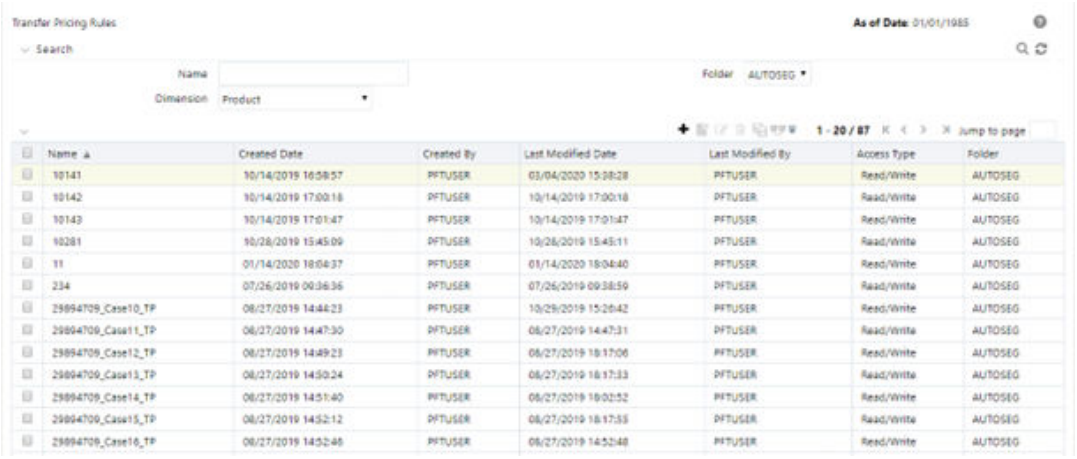
- Update, copy, delete, or Run existing Rules.
- Define methodologies for products or define other processing assumptions.

To search the Rules:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Assumption Specification**, and then select **Transfer Pricing Rules** to display the Transfer Pricing Rules Summary page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 13-1 Transfer Pricing Rules Summary page



2. Search for the Rule, as follows:
   a. Select the folder in which the Rule is stored.
   b. (Optional) Enter the name of the Rule.
   c. Click **Search**.
      Only Rules that match the search criteria are displayed.

13.3 Create Rules

You create a Rule to specify the way you want a particular task or business process to be carried out by the application.

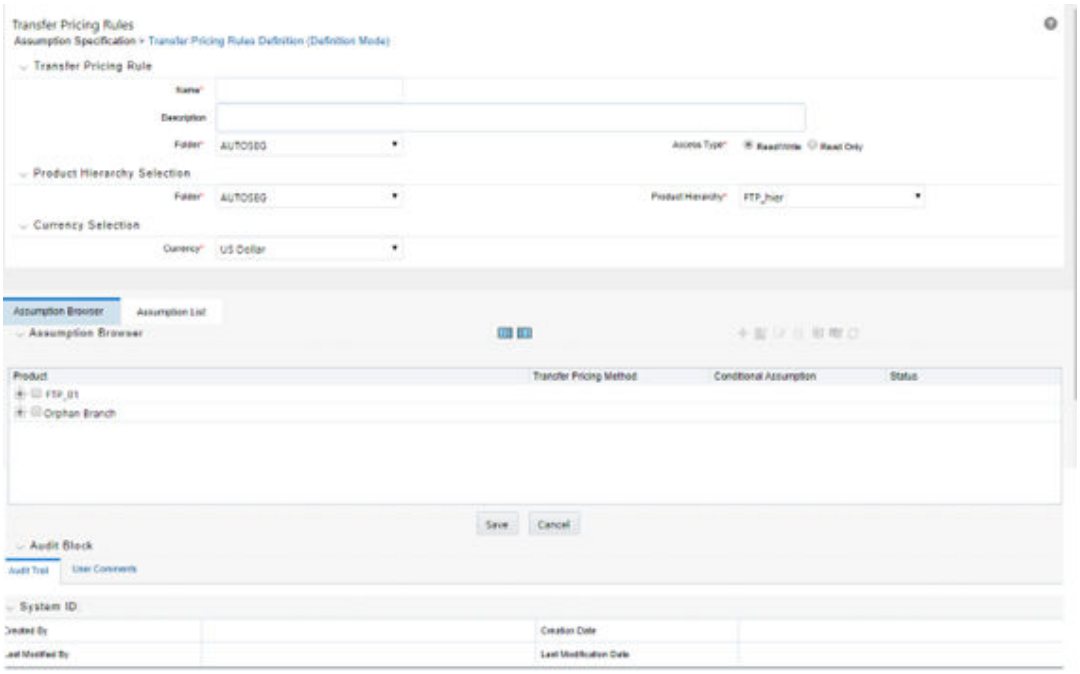
Creating a Rule is a process, in which you specify the properties for the Rule itself.

To create a rule:

1. Navigate to the Summary page of the rule you want to create.
2. Click the **Add** icon to display the rule Definition page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 13-2 Transfer Pricing Rule - Definition Mode



3. Enter a name for the Rule.
   The name of a Rule must be unique within the selected folder for each Rule type.
4. Select the folder in which you want to store the Rule.
5. (Optional) Enter a description for the Rule.
6. Select the required access for other users.
7. Click **Apply** or **Save**, depending on the Rule type.
8. Specify any other properties or options that may apply to the Rule that you are creating.

13.4 View and Edit Rules

You can view existing rules, and you can edit existing Rules, provided you have read and write privileges.

To view a rule:

1. Navigate to the Summary page of the rule you want to update.
2. Search for a rule.
   The details of the selected rule are displayed.

To edit the rule:

1. Select the appropriate rule and click the **Edit** icon to open the rule you want to update.
2. Update the Name or Description.
3. Click **Apply** or **Save** depending on the rule type.

Copyright © 2011, 2026, Oracle and/or its affiliates.

13.5 Copy Rules

You can copy rules to avoid having to enter data multiple times. This saves time and effort and also reduces mistakes.

To copy rules:

1. Navigate to the Summary page of the rule you want to copy.
2. Search for a rule.
3. Select the appropriate rule and click the **Copy** icon corresponding to the rule that you want to duplicate.
4. Select a folder.
5. Enter a unique name for the new rule.
6. (Optional) Enter a brief description of the rule.
7. Select the access type.
8. Click **Save**.

13.6 Delete Rules

You can delete rules that are no longer needed. Once deleted, a rule cannot be retrieved.

The following are the restrictions on deleting Rules:

- You cannot delete rules if you have only Read privileges. Only users with Read or Write privileges and rule owners can delete rules.
- You cannot delete a rule that has a dependency.

To delete a rule:

1. 1. Navigate to the Summary page of the rule you want to delete.
2. Search for a rule.
3. Select the appropriate rule and click the **Delete** icon.

13.7 Dependency Checking

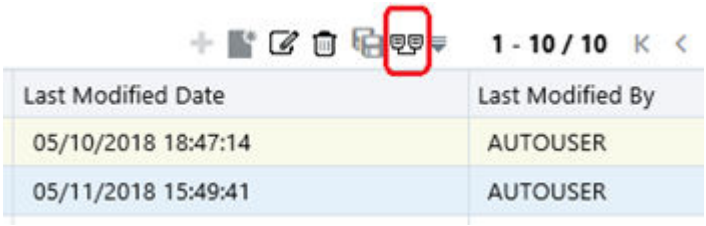
You can check dependencies for rules to know where a particular Rule, Dimension, Member, Attribute, IRC has been used. In addition, this prevents accidental deletion of rules having dependencies.

To check the dependencies for rules:

1. Navigate to the Summary page of the rule you want to check dependencies.
2. Search for a rule.
3. Select the appropriate Rule(s) and click check dependencies corresponding to the Rule that you want to check for.
   You can select more than one rule at a time to check dependencies.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 13-3 Transfer Pricing Summary page - Dependency Check



The Dependency Information window opens containing the information: Child Object Name, Child Object Type, Folder, Parent Object Name, Parent Object Type, and Folder.

Figure 13-4 Transfer Pricing Rule - Dependency Information



If a Product Characteristics rule , ‘A’ which has been used in a static deterministic process ‘P’, is checked for dependencies, then the following information is displayed in the Dependency Information window.

- Child Object Name –A
- Child Object Type – Product Characteristics
- Folder – The folder name in which A resides
- Parent Object Name - P
- Parent Object Type – Static Deterministic Process
- Folder - The folder name in which P resides

> **Note**
>
> You cannot delete a child object, which has dependencies. You will get a message: Dependencies found. Cannot delete.
> To delete the Child Object, you must first delete the outermost Parent Object which uses the Child Object.

Copyright © 2011, 2026, Oracle and/or its affiliates.

14

Application Preferences

Application Preferences allow both Administrators and End Users to establish default values and to manage other core application parameters that affect the way business rules are created and the way Transfer Pricing Processes are Run.

The procedure for working with and managing Application Preferences includes updating the FTP Application Preferences.

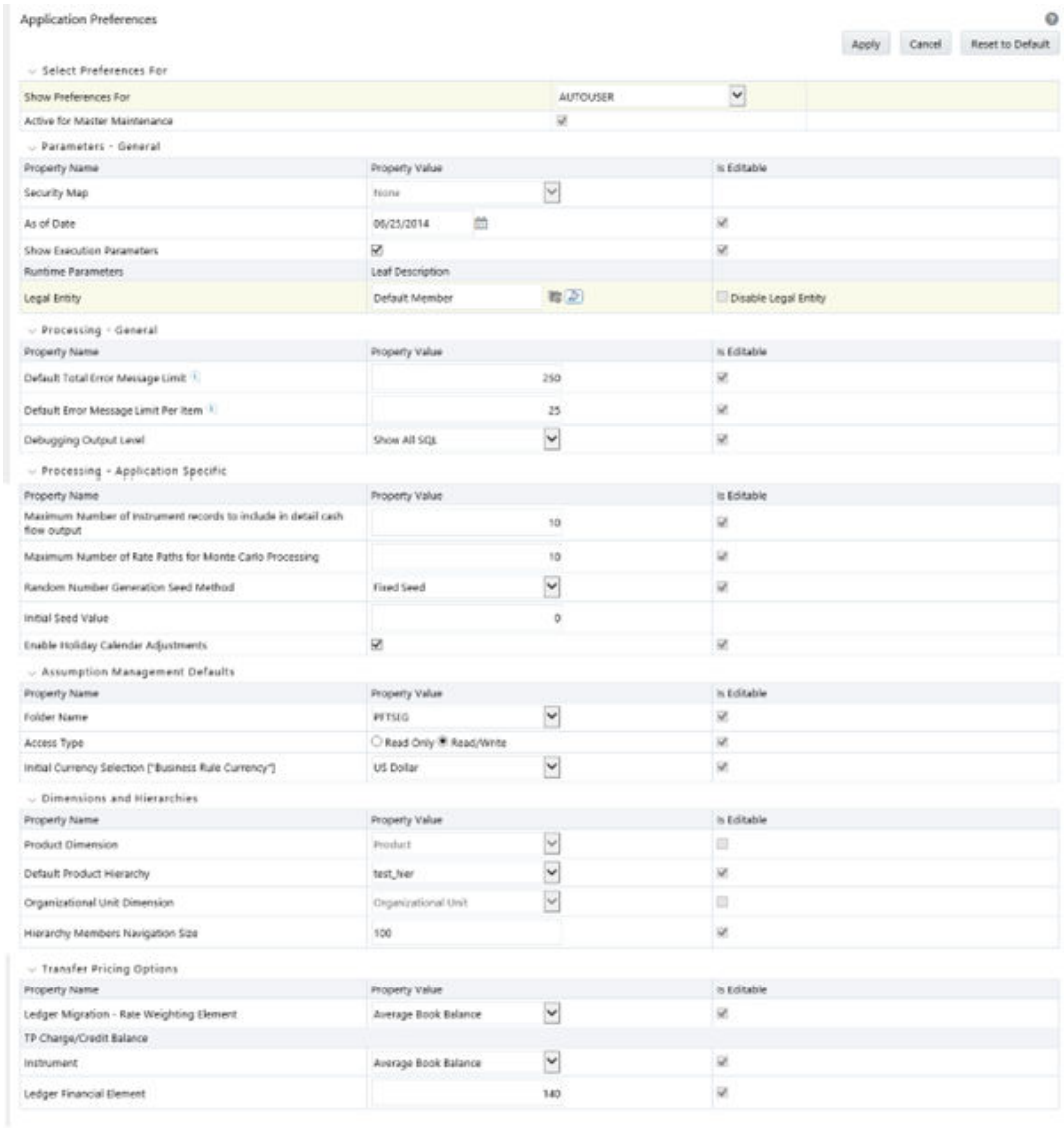
Updating FTP Application Preferences is a one-step process. You navigate to the Application Preferences screen and you define your preferences.

To update the FTP Application Preferences:

1. From the LHS menu, select **Funds Transfer Pricing**, and then select **Application Preferences** to display the Application Preferences page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 14-1 Application Preferences page



The following table describes the key terms used for this procedure.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 Fields and Descriptions from the Application Preferences screen

Term	Description
Show Preferences For	There are two modes in which you can access Application Preferences: • **Administrator**: If the user has Administrator privileges, he can define preferences for the “All User” group and for his own personal account, which may be the same or different from the “All User” settings. The Administrator can also designate the “All User” preferences as Editable or Non-editable on a row-by-row basis. If the individual preference is checked as “Is Editable”, then End Users can update or override the Administrator’s default value for their own individual account. If the “is editable” box is not checked, then End Users are not able to change the default for their own account. • **End User**: If the user does not have administrator privileges, then certain preference items may are pre-set by the administrator and the user may not be allowed to change the value. All Application Preference settings are displayed, regardless of access privilege.
Active for Master Maintenance	If a user has more than one application available on their Left-Hand menu, then they will need to designate Application Preferences from one of their applications as “Active for Master Maintenance”. Items within Master Maintenance based on this selection will reference certain preferences such as the default folder and read or write access setting.
Security Map	This option allows you to select an existing security map. Security Maps can be used to control the dimensions, dimension members each user can access when building, and executing rules. For more information on Security Mapping, see the OFS Analytical Applications Infrastructure User Guide.
As-of-Date	All processes reference this date at run time to determine the data to include in the process. The As-of-Date value you set in Application Preferences applies to interactive job execution (i.e., when you choose to execute a process directly from a Summary Screen). For batch processing, the As-of-Date is an input parameter.
Show Execution Parameters	If this option is selected, a pop-up window appears whenever you execute a process interactively from a Summary Screen. Within this pop-up window, you may confirm or modify your run execution parameters (As-of-Date and Legal Entity).

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 (Cont.) Fields and Descriptions from the Application Preferences screen

Term	Description
Legal Entity	Similar to As-of-Date, all processes reference Legal Entity at run time to determine the data to include in the process. The value of the Legal Entity you set in Application Preferences applies to interactive job execution (i.e., when you choose to execute a process directly from a Summary screen). For batch processing, Legal Entity is an input parameter. **NOTE**: Legal Entity is designed to support implementations that require multi-entity or multi-tenant functionality. If your implementation does not require this functionality, you may utilize the Default Legal Entity in all your processes. No additional parameter is required for the Legal Entity for command-line execution. EPM Engines read the default Legal Entity from the Application preference value saved for the User who is executing from the command line. The default legal entity is a per-user preference setting. To set this, the following steps are required: **a.** Create a security-map in AAI containing the legal-entity hierarchy as an app-admin. **b.** Set the default security-map in user preferences. **c.** Navigate to the "execution parameters" block in user-preferences and choose the default legal-entity member. If you do not want to use the Multi-Entity feature with Security, you need to save the Application preference for the user executing the batch. The usual choice for Legal Entity would be Default Member in Application preferences. Default implies -1 code. The default value for the Legal Entity dimension column in all instruments and ledger table is -1.
Disable Legal Entity	If the "Disable Legal Entity" check box in the application preferences screen is checked, then the Legal Entity will no longer be a Run-time parameter. On disabling Legal Entity, the user should edit the already defined Variable Allocation rules and modify the value of the Legal Entity dimension in all applicable tabs (Source, Driver, and Outputs) in the allocation specification screen. This is an optional parameter.
Default Total Error Message Limit	Transfer Pricing Processes log error details into the FSI_PROCESS_ERRORS table. This parameter defines the limit on the total number of errors that will be logged for any TP Process.
Default Error Message Limit Per Item	This parameter defines the total number of errors that will be logged for a given type of error.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 (Cont.) Fields and Descriptions from the Application Preferences screen

Term	Description
Debugging Output Level	The debugging output level determines the amount of SQL that will be written to the processing log. There are three levels available: • **Do not output SQL**: A log file will not be created. • **Output Significant Calculation SQL**: Log file is created and will contain those SQLs that are tagged as significant. • **Output All SQL**: Log file is created and will contain all the SQL that the engines execute. The log files can be accessed by the system administrator in the following location on the server: $FIC_HOME/ficdb/log/FusionApps/ folder The file names will be prefixed with the application initials and will also contain the unique batch run id of the execution request. for example: ofstp.<batch-run-id>.log
Maximum Number of Instrument Records to Include in Detail Cash Flow Output	This parameter allows administrators to define the maximum number of instrument records that any user can select within a process for outputting detailed cash flows. In Funds Transfer Pricing, the maximum value is 10,000. It is recommended, however, that this value be set to 100 or less.
Maximum Number of Rate Paths for Monte Carlo Processing	This parameter allows administrators to limit the number of rate paths that can be selected by a user when running a Monte Carlo based process. The maximum value is 2,000.
Random Number Generation Seed Method	Determines the type of random number generation seed method for Monte Carlo processing. Selections include Fixed Seed (default) and Variable Seed.
Initial Seed Value	The Initial seed value input is available when the Seed Value Method is Fixed Seed. The default value is zero.
Option Cost Precision Factor	The Option Cost calculator - Static Spread value is calculated using the Newton-Raphson algorithm. Users can control the convergence speed of the algorithm by adjusting this value. Precision increases as this value decreases, but processing time also increases as the value decreases. The default value is one.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 (Cont.) Fields and Descriptions from the Application Preferences screen

Term	Description
Enable Holiday Calendar Adjustments	Select this option to enable the Holiday Calendar Adjustment capability for the FTP Application. If this option is not selected, the TP Engine ignores all Holiday Calendar information, including instrument level inputs and assumption rule level inputs. The logic for applying Holiday Calendar assumptions is as follows: • If Application Preferences - 'Enable Holiday Calendar Adjustments' checkbox is on, then the CFE will handle these Holiday Calendar assumptions based on the Account Level values first. • If Application Preferences – 'Enable Holiday Calendar Adjustments' checkbox is on, and if Holiday Calendar inputs are not defined at the Account Level, then the CFE will refer to the Product/Currency assumptions (TP rule and Adjustment rule). • If Application Preferences – 'Enable Holiday Calendar Adjustments' checkbox is on, and if Holiday Calendar inputs are defined at the Account Level AND Product or Currency Assumption level, then the CFE refers to the Account level inputs. • If Application Preferences – 'Enable Holiday Calendar Adjustments' checkbox is off, then no Holiday Calendar assumptions are applied.
Folder Name	This parameter allows you to define the default folder selection. The folder selection for all rule types will be defaulted to this selection within the Summary Page Search Screen and when creating a new rule. This selection acts as the starting value for convenience only and users can change to any other available value at their discretion.
Access Type	This parameter allows you to set the default access typesetting. Selections include Read / Write and Read Only. This selection acts as the starting value for convenience only and users can change at their discretion.
Initial Currency Selection ("Business Rule Currency")	This parameter allows you to select the starting currency to be displayed within all business rules. This selection is made for convenience and can be changed within all business rules at the user's discretion.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 (Cont.) Fields and Descriptions from the Application Preferences screen

Term	Description
Product Dimension	Transfer Pricing requires users to declare one of the “Product” dimensions as the TP Product dimension. The model is seeded with three possible selections: • Product • Common COA • GL Account Users can also add user-defined product dimensions, which would also appear in the above list. Transfer Pricing business rules are based on the Product dimension selected here. The suggested default is the “Product” dimension.
Default Product Hierarchy	The list of values for the Default Product Hierarchy is based on the Default Product Dimension selection. The hierarchy selected here will be the default hierarchy selection in all business rules that support node-level assumptions. This selection acts as the starting value for convenience only and users can change at their discretion within each business rule.
Organizational Unit Dimension	Reserved for future release.
Hierarchy Members Navigation Size	This parameter allows you to specify the maximum number of members that a parent node within a Hierarchy or Assumption Browser can show at a given time. When you expand a branch in a Hierarchy or Assumption Browser and the number of members in that branch exceeds the specified Navigation Size, it will provide "More" and "Previous" options to enable you to navigate through the member list. Recommended values for Navigation Size are 50 to 100. Higher value settings could impact screen refresh performance. The UI allows you to enter a value up to 10000.
Ledger Migration – Rate Weighting Element	Select the instrument table balance to use for weighting the rates during the migration process. The following options are available: Average Book Balance, Ending Book Balance, or Custom Balance. If “Custom Balance” is selected, the user is presented with a list of Balance type columns to use as the weighting element. The list of available “Custom Balance” columns is read from the “Portfolio” table classification list.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 14-1 (Cont.) Fields and Descriptions from the Application Preferences screen

Term	Description
TP Charge / Credit Balance	Select the Balance to use for calculating the Charge/Credit amount. Select from Ending Book Balance or Average Book Balance when using the "Ledger" based migration option. For calculating instrument-level charge/credit amounts, you may also choose the Custom Balance option. The formula used is: Rate x TP Charge / Credit Balance x Accrual Basis Available selections include Ending Balance, Average Balance, and Custom Balance. If Custom Balance is selected, the user is presented with a list of Balance type columns to choose from. The list of available “Custom Balance” columns is read from the “Portfolio” table classification list. The user should be aware of the Transfer Pricing Migration methodology used before making this selection. The custom balance option only applies to the Instrument Charge / Credit method and does not apply to Ledger Migration.

2. Input the values for all the line items.

> **Note**
>
> If you are the application administrator, define default values for the “All User” group by making the appropriate selection from the drop list at the top of the page. Pay particular attention to the “Is Editable” status and determine which items require administrative control and which items non-administrative users will be able to set for themselves.

3. Select **Apply** to confirm changes.

Copyright © 2011, 2026, Oracle and/or its affiliates.

15

Funds Transfer Pricing Configuration

The FTP Configuration UIs allow both Administrators and End Users to establish default values and to manage other core configuration parameters.

Topics:

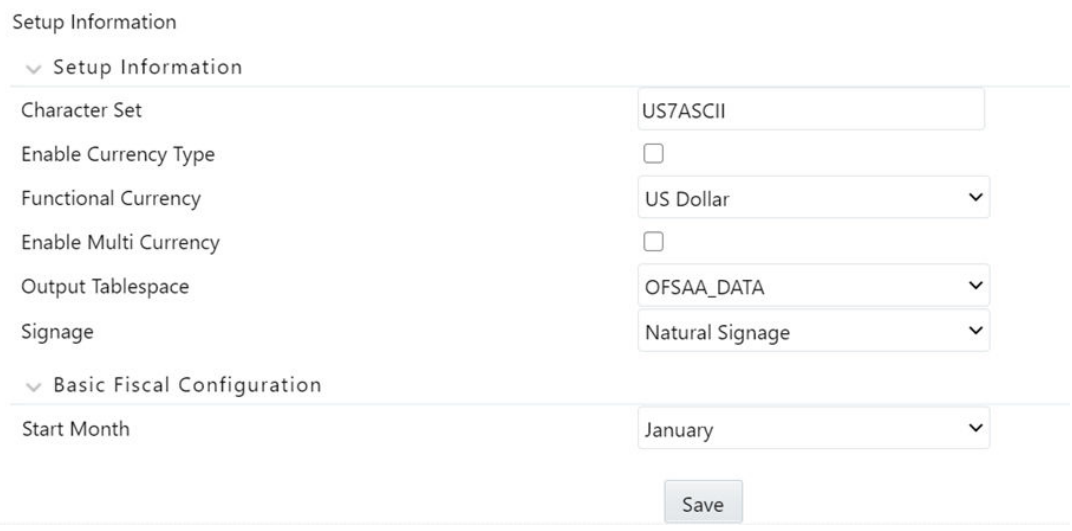
- Setup Information
- Parallel Execution Configuration
- Setup Parameters
- Management Ledger Configuration
- Setup Information
  To configure the Setup Information:
- Parallel Execution Configuration
  To configure Parallel Execution, follow these steps:
- Setup Parameters
  To configure Setup Parameters:
- Management Ledger Configuration
  To configure the Management Ledger:

15.1 Setup Information

To configure the Setup Information:

1. From the LHS menu, select **Funds Transfer Pricing**, select **Configuration**, and then select **Setup Information** to display the Setup Information page.

Figure 15-1 Setup Information page



Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Select the **Enable Currency** checkbox to enable the currency type.
3. Select the relevant **Functional Currency**. Accounts and instruments can be defined across various currencies, but to consolidate the accounts' balances or charges at multiple hierarchy levels, a common functional currency is required which can be set here.
4. Select the **Enable Multi Currency** checkbox to enable multi-currency. Not applicable for FTP.
   The Output Tablespace is not applicable to the FTP application.
5. Select a relevant option for **Signage** from the following options:
   - Natural Signage
   - GAAP Signage
   - Reverse GAAP Signage

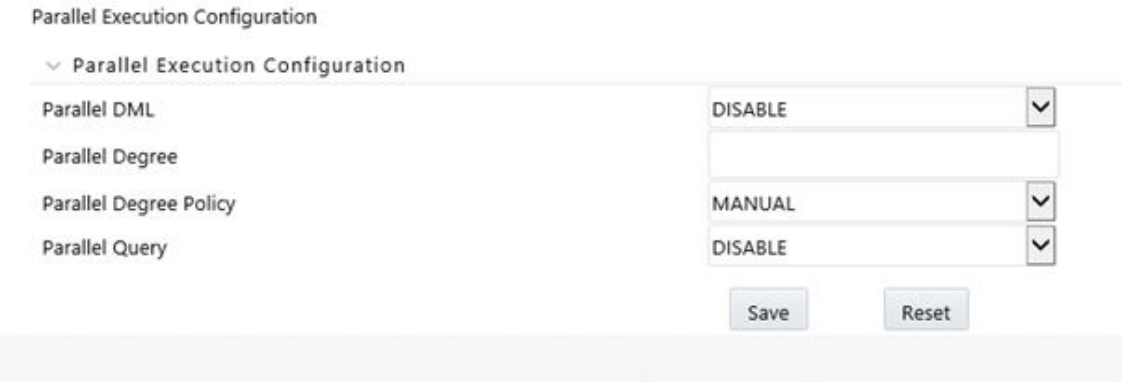
   This is an information only setting describing how balances are carried.
6. Select a **Start Month** of the fiscal year. This helps to define the start of a fiscal year. A fiscal year is a one-year period that customers use for financial reporting and budgeting. A fiscal year is most commonly used for accounting purposes to prepare financial statements. Although a fiscal year normally starts on January 1 and ends on December 31, you can change it as per your local norms using this field.
7. Click **Save**.

15.2 Parallel Execution Configuration

To configure Parallel Execution, follow these steps:

From the LHS menu, select Funds Transfer Pricing, select Configuration, and then select the Parallel Execution Configuration to display the Parallel Execution Configuration page.

Figure 15-2 Parallel Execution Configuration page



> **Note**
>
> Parallel Execution Configuration does not apply to FTP Application. These settings will impact only the OFS Profitability Management Application.

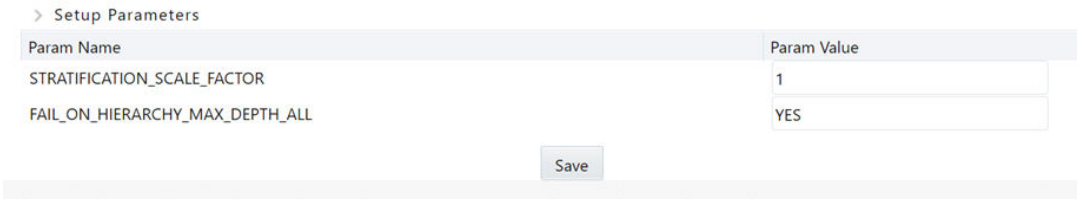
Copyright © 2011, 2026, Oracle and/or its affiliates.

15.3 Setup Parameters

To configure Setup Parameters:

1. From the LHS menu, select **Funds Transfer Pricing**, select **Configuration**, and then select **Setup Parameters** to display the Setup Parameters page.

Figure 15-3 FTP Configuration - Setup Parameters page



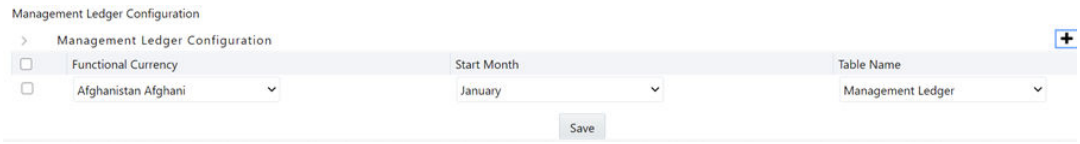
2. Enter the following details:
   - **STRATIFICATION_SCALE_FACTOR**: This parameter would help to scale the large numbers like aggregate balances in case they are beyond the range defined for a particular column.
   - **FAIL_ON_HIERARCHY_MAX_DEPTH_ALL**: If this is set to True, the Hierarchy load will fail if there is any member more than beyond the permitted hierarchy depth.
   - Click **Save**.

15.4 Management Ledger Configuration

To configure the Management Ledger:

1. From the LHS menu, select Funds Transfer Pricing, select Configuration, and then select the Management Ledger Configuration to display the Management Ledger Configuration page.

Figure 15-4 FTP Configuration - Management Ledger Configuration page



2. Select the following details:
   - Functional Currency: This helps to define a Functional Currency for the ledger table. Accounts and instruments can be defined across various currencies, but to consolidate the accounts at multiple hierarchy levels, across units, a common Functional Currency is required which you can set here.
   - Start Month: The start month of the fiscal year. You can choose a ledger table and define the corresponding start month or functional currency.
   - Table Name: The name of the table.
3. Click Save.

Copyright © 2011, 2026, Oracle and/or its affiliates.

For more information on Functional Currency, see Functional Currency.

Copyright © 2011, 2026, Oracle and/or its affiliates.

16

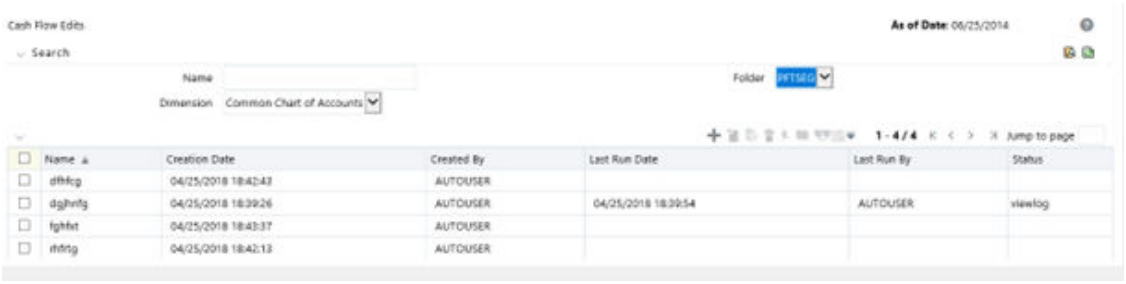
Cash Flow Edits

Cash Flow Edit processes allow you to verify the accuracy and check the completeness of your Instrument table data. The procedure for working with and managing a Cash Flow Edit process is similar to that of other Oracle Funds Transfer Pricing and Oracle Asset Liability Management Processes.

It includes the following steps:

- Searching for Cash Flow Edit Processes
- Viewing and Updating Cash Flow Edit Processes
- Copying Cash Flow Edit Processes
- Deleting Cash Flow Edit Processes
- Check Dependencies in the Cash Flow Edit Processes
- Refresh the Cash Flow Edit Summary Page

Figure 16-1 Cash Flow Edits Summary page



You should create and run Cash Flow Edit Processes on your Instrument table data before you submit Cash Flow Engine based rules for processing. For more information, see the OFS Cash Flow Engine Reference Guide.

- Creating Cash Flow Edit Processes
  Creating a Cash Flow Edit Process is a one-step process. You define both the attributes that uniquely describe a particular Cash Flow Edit Process and the data to be validated or cleansed by that process on the Create Cash Flow Edit Process page.
- Executing Cash Flow Edit Processes
  You execute a Cash Flow Edit Process to check the accuracy and the completeness of your Instrument table data. When you run in Preview mode, you can view the results of running a Cash Flow Edits Process by querying the FSI_PROCESS_ERRORS table for generated errors before the system updates the underlying records in the Instrument tables.

Copyright © 2011, 2026, Oracle and/or its affiliates.

16.1 Creating Cash Flow Edit Processes

Creating a Cash Flow Edit Process is a one-step process. You define both the attributes that uniquely describe a particular Cash Flow Edit Process and the data to be validated or cleansed by that process on the Create Cash Flow Edit Process page.

To create Cash Flow Edit Processes:

1. Navigate to the Cash Flow Edits Summary page.
   The following table describes the key terms used for this procedure.

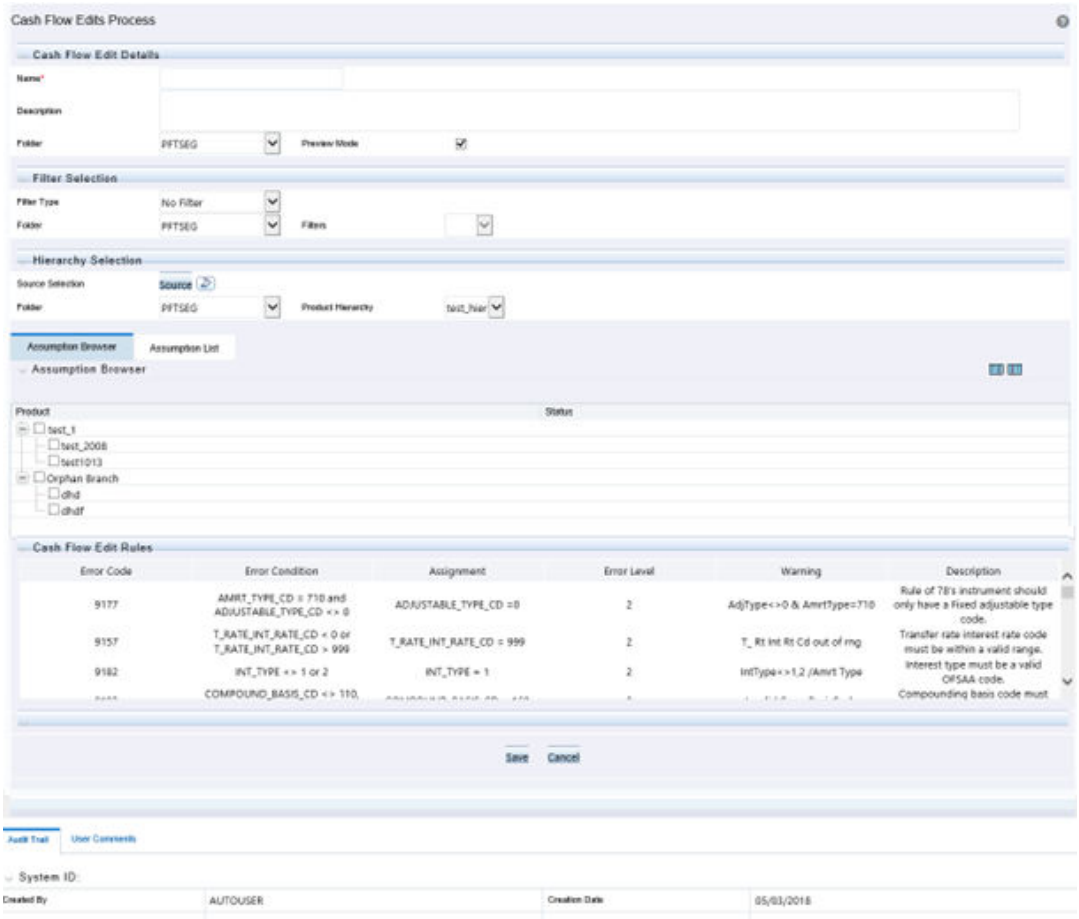
Table 16-1 Terms used in the Cash Flow Edit Summary page

Term	Description
Filters	One of the two components that determine the data that will be cleansed by Cash Flow Edit processes. This field allows you to select a subset of data for processing by selecting a Filter that was previously created. Its default value is "No Filter".
Source Selection	One of the two components that determine the data that will be cleansed by Cash Flow Edit processes. This field allows you to select the Instrument tables that need to be included in a Cash Flow Edit process. Alternatively, you can select a hierarchy and then select the desired product members that will be included in the process. Note that the Source selection and Hierarchy selection options are mutually exclusive. You can define your dataset by selecting one or the other, but not a combination of both.
Preview Mode	Selecting this checkbox allows you to view the results of running a Cash Flow Edit Process before the system updates the underlying records in the Instrument tables. The default value is checked.
Source Selection: Available Tables	When the "Source Selection" option is made, you are presented with two Shuttle Control windows that contain the names of the Instrument Tables available for inclusion during a Cash Flow Edit process.
Source Selection: 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.
Cash Flow Edit Rules	The Cash Flow Edit Rule section provides a summary of all seeded cash flow edit rules. Users can refer to this list to understand the validations that are applied to the cash flow fields on the instrument records and view the default values that are applied when errors are found.

2. Click **Add** to display the Create Cash Flow Edits Process page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 16-2 Cash Flow Edits Process page



3. Complete standard steps for this procedure. For the details, see Creating Transfer Pricing Rules.

> **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 selecting Instrument tables or selecting product members from the hierarchy, the Process will be saved but no data would be selected for cleansing.

4. (Optional) Select a **Filter**.
5. Select the Instrument tables or use a hierarchy to make Product member selections.

> **Note**
>
> Use the Source Selection: Shuttle Control to select the Instrument tables that you want to include in the Cash Flow Edit process. You can move Instrument 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

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 Process, the system shows them the next time you open the rule.

A table name shown in the Selected Tables list does not appear in the Available Tables.

6. Click **Save**. The Cash Flow Edits process is saved and the Summary Page is displayed.

16.2 Executing Cash Flow Edit Processes

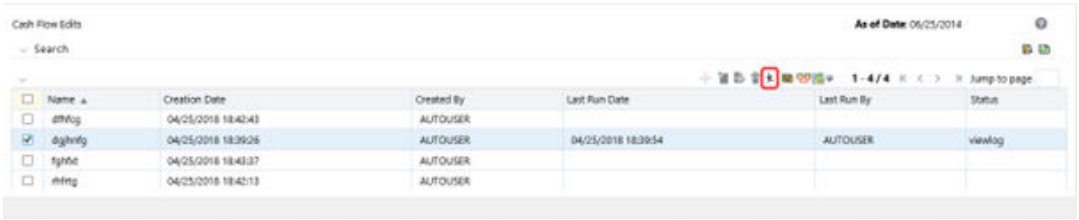
You execute a Cash Flow Edit Process to check the accuracy and the completeness of your Instrument table data. When you run in Preview mode, you can view the results of running a Cash Flow Edits Process by querying the FSI_PROCESS_ERRORS table for generated errors before the system updates the underlying records in the Instrument tables.

You must define the Predefined Rules before executing the Cash Flow Edit processes.

To execute the Cash Flow Edit Processes:

1. Navigate to the Cash Flow Edits Summary page.

Figure 16-3 Cash Flow Edits Summary page



2. Search for a rule.
3. Select a rule and click the **Run** icon to execute the selected process. The Cash Flow Edits Run confirmation 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 Instrument tables, provided you selected Preview Mode while defining it. If Run in Preview Mode, query the FSI_PROCESS_ERRORS table for any generated errors.

You can also execute the Cash Flow Edit process by navigating to **Common Object Maintenance**, select **Operations**.

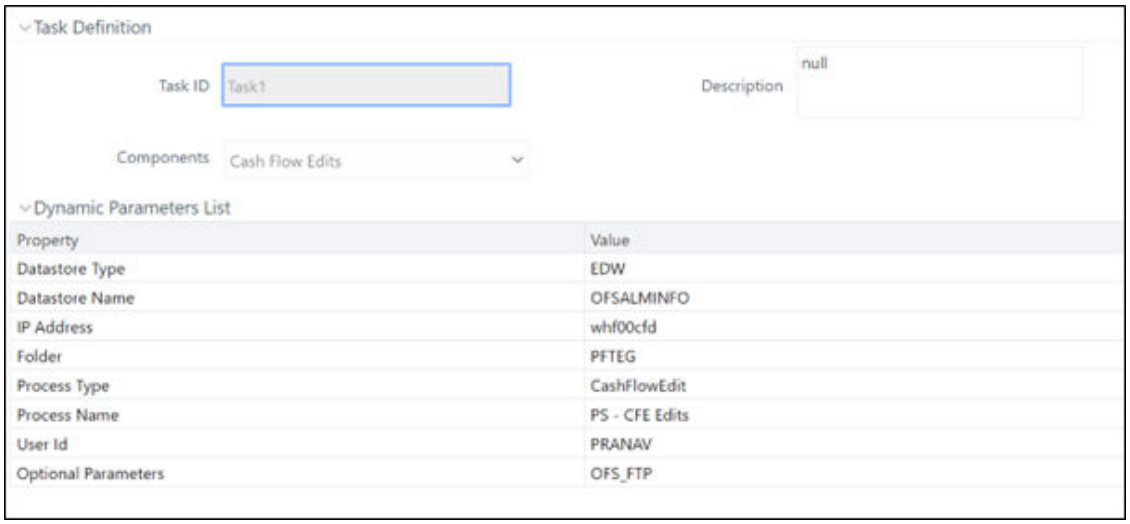
To execute the Cash Flow Process, create a new batch and specify the following parameters:

- Datastore Type: Select appropriate datastore from the list.
- Datastore Name: Select appropriate name from the list.
- IP address: Select the IP address from the list.
- Folder: Select appropriate name from the list.
- Process Type: Select “CashFlowEdit” from the list.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Process Name: Select process name from the list.
- Optional Parameters: OFS_FTP.

Figure 16-4 Task Definition



As of Date selection can be selected when the batch is executed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

17

Cash Flow Edits New

This module discusses the procedure for validating and cleansing your Instrument table data before you process it to generate Cash Flow based results.

This chapter covers the following topics:

- Overview of Cash Flow Edit Processes
- Creating Cash Flow Edit Processes
- Executing Cash Flow Edit Processes
- Overview of Cash Flow Edit Processes
  Cash Flow Edit Processes allow you to verify the accuracy and check the completeness of your Instrument table data.
- Creating Cash Flow Edit Process
  Creating a Cash Flow Edit Process is a one-step process. You define both the attributes that uniquely describe a particular Cash Flow Edit Process and the data to be validated or cleansed by that process on the Create Cash Flow Edit Process Page.
- Executing Cash Flow Edit Process
  Execute a Cash Flow Edit process to check the accuracy and the completeness of your Instrument table data. When run in Preview Mode, you can view the results of running a Cash Flow Edits process by querying the FSI_PROCESS_ERRORS table for generated errors before the system updates the underlying records in the Instrument tables.

17.1 Overview of Cash Flow Edit Processes

Cash Flow Edit Processes allow you to verify the accuracy and check the completeness of your Instrument table data.

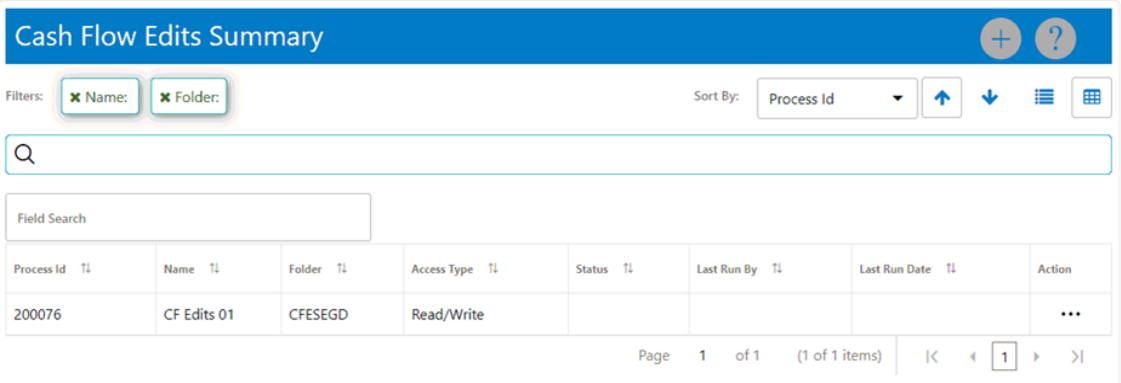
> **Note**
>
> Cash Flow Edits do not support the Derivatives Tables.

The procedure for working with and managing a Cash Flow Edit Process includes the following steps:

- Searching for Cash Flow Edit processes: For more information, see the Search for Rules Section.
- Creating Cash Flow Edit processes: For more information, see the Create Rules Section.
- Viewing and Editing Cash Flow Edit processes: For more information, see the View and Edit Rules Section.
- Deleting Cash Flow Edit processes: For more information, see the Delete Rules Section.
- Check Dependencies in the Cash Flow Edit Processes.
- Refresh the Cash Flow Edit Summary page.

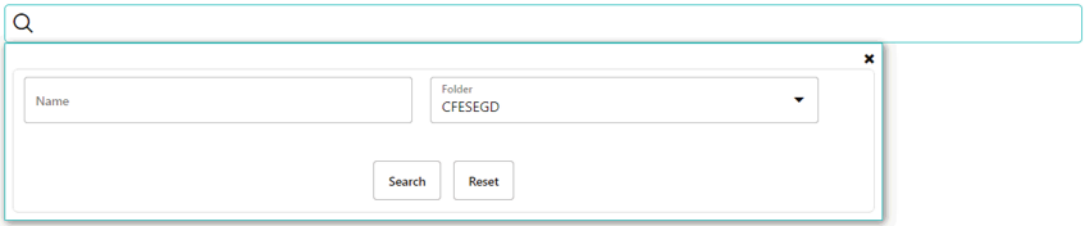
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 17-1 Cash Flow Edit Summary page



The definitions based on the search criteria are listed under the List of Cash Flow Edits. This is the search section that contains multiple parameters. You can specify one or multiple search criteria in this section. When you click Search, depending upon the search criteria, this filters and displays the relevant search combination parameters under the Cash Flow Edits Summary as a list.

Figure 17-2 Search page



The Cash Flow Edit Summary Window displays the following columns:

- **Process ID**: Displays the Process ID of Cash Flow Edit Rule.
- **Name**: Displays the Cash Flow Edit rule's short name.
- **Folder**: Displays the Folder name where Cash Flow Edit Rule is saved.
- **Access Type**: Displays the access type of rule. It can be Read-Only or Read/Write.
- **Status**: Displays the status of the rule.
- **Last Run By**: Displays the Name of the user who last run the Cash Flow Edit Rule.
- **Last Run Date**: Displays the Date and Time when Cash Flow Edit was run last.
- **Action**: Displays the list of actions that can be performed on the Cash Flow Edit Rule. For more information, see Cash Flow Edit Rule – Icons and Descriptions.

The Action column on Cash Flow Edits Summary Page offers several actions that allow you to perform different functions. The following actions are available for the Cash Flow Edit Rule.

- **Add**: Click the Add icon to build a new Cash Flow Edit Rule.
- **View**: Click the Action column and select View/Edit to view or edit the contents of a Cash Flow Edit Rule in read/write format.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Save As**: Click the Action column and select Save As to create a copy of an existing Cash Flow Edit Rule.
- **Delete**: Click the Action column and select Delete to delete an existing Cash Flow Edit Rule.
- **Execute**: Click the Action column and select Execute to execute an existing Cash Flow Edit Rule. After clicking Execute, the Run Parameter Execution window is displayed. Select As of Date (Execution Date) and Legal Entity, and then click Run. For more information, see the Executing Cash Flow Edit Processes Section.
- **Execute Details**: Click the Action column and select Execute Details to view execution details of the Cash Flow Edit Rule.

You must create and run Cash Flow Edit Processes on your Instrument table data before you submit Cash Flow Engine-based rules for processing.

17.2 Creating Cash Flow Edit Process

Creating a Cash Flow Edit Process is a one-step process. You define both the attributes that uniquely describe a particular Cash Flow Edit Process and the data to be validated or cleansed by that process on the Create Cash Flow Edit Process Page.

To create a Cash Flow Edit process:

1. Navigate to the Cash Flow Edits Summary page.
2. Click **Add**. The Create Cash Flow Edits Process Page is displayed.

Figure 17-3 Cash Flow Edit Process of CFE Processing



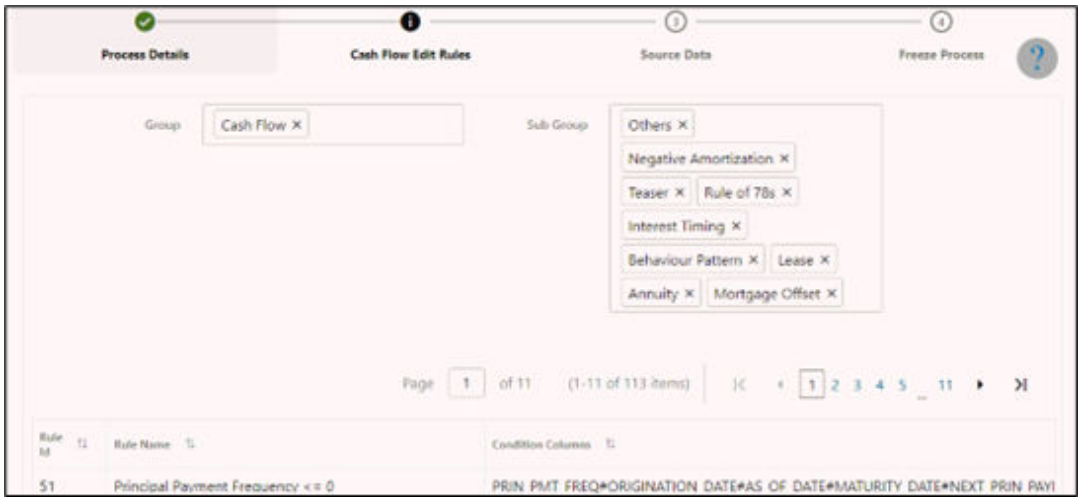
3. Enter the following process details:
   - **Name**: Enter the name of the Cash Flow Edit Rule.
   - **Description**: Enter the description of the Cash Flow Edit Rule.
   - **Preview Mode**: Selecting this parameter allows you to view the results of running a Cash Flow Edit Process before the system updates the underlying records in the Instrument tables. The default value of this parameter is checked.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Folder**: Enter the Folder details where Cash Flow Edit Rule needs to be saved.
- **Access Type**: Select the Access Type as Read-Only or Read/Write.

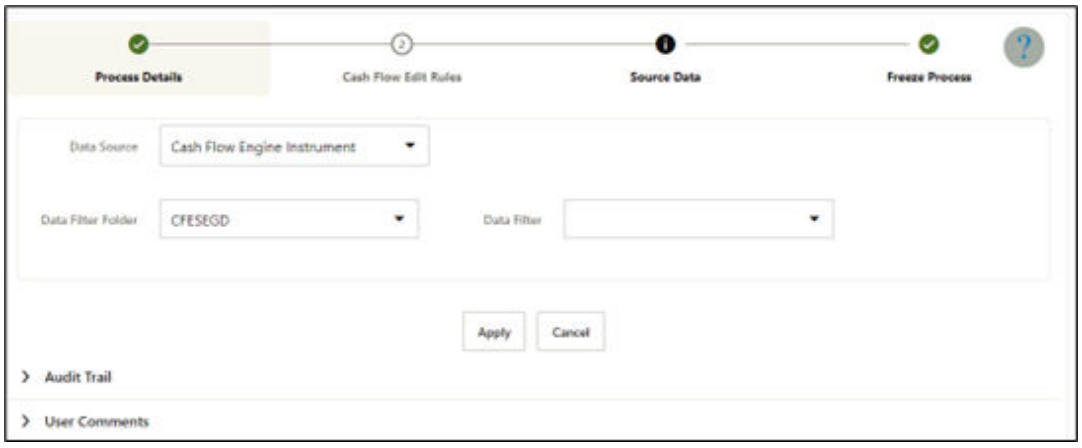
4. Click **Apply** to navigate to Cash Flow Edit Rules Section.

Figure 17-4 Cash Flow Edit Rules



5. Enter the following Cash Flow Edit details:
   - **Group**: Select the group of Instrument tables of Cash Flow Edit Rule that you want to include in the Cash Flow Edit process. For example: Cash Flow.
   - **Sub-Group**: Select the Instrument table of Cash Flow Edit Rule that you want to include in the Cash Flow Edit process. The value of this field varies based on the selected Group. The list of Rule IDs with conditions is displayed.
6. Click **Apply** to navigate to the Source Data Section.

Figure 17-5 Source Data Tab



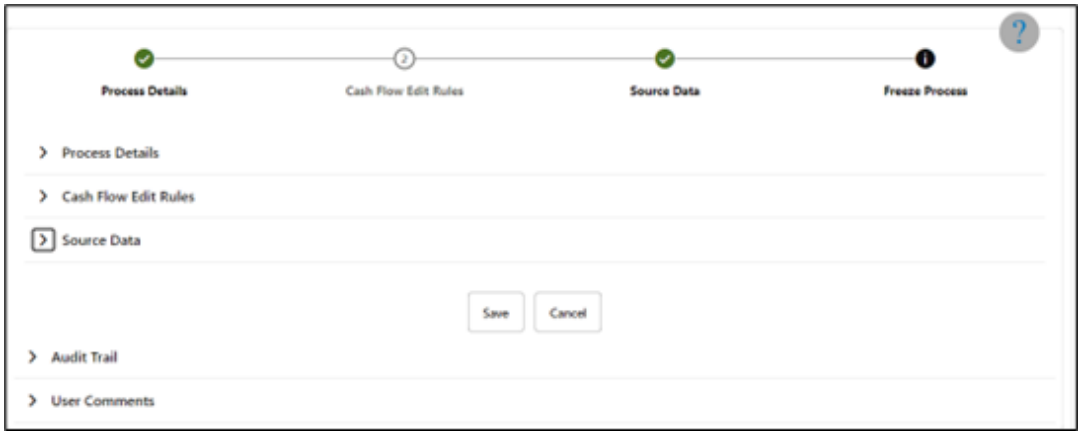
7. Enter the following Source Data details:
   - **Data Source**: This field allows you to select the Instrument tables that must be included in a Cash Flow Edit Process.
   - **Data Filter Folder**: Select the Data Filter Folder.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Data Filter**: This field allows you to select a subset of data for processing by selecting a Filter that was previously created.

8. Click **Apply** to navigate to the Freeze Process section. Click **Save** on the Freeze Process window after verifying all the details.

Figure 17-6 Freeze Process tab



The Cash Flow Edits process is saved and the Summary page is displayed.

17.3 Executing Cash Flow Edit Process

Execute a Cash Flow Edit process to check the accuracy and the completeness of your Instrument table data. When run in Preview Mode, you can view the results of running a Cash Flow Edits process by querying the FSI_PROCESS_ERRORS table for generated errors before the system updates the underlying records in the Instrument tables.

The following list describes the key terms used for this procedure.

You can execute the Cash Flow Edit process using the following methods:

- Cash Flow Edit UI
- Batch Maintenance
- Using Cash Flow UI
  To execute the cash flow edits process
- Batch Based Procedure
  Cash Flow Edit Processes can also be executed as Batch Processes. The following approaches are available:
- Using the Command Line
  This topic describes the execution of the Enhanced Cash Flow Edits process for Funds Transfer Pricing (FTP) from the command line in Oracle Financial Services Analytical Applications (OFSAA) Release 8.1. Oracle recommends executing the Enhanced Cash Flow Edits process through the user interface; command-line execution is intended for advanced or administrative use cases.
- View Cash Flow Edit Process Log
  To view the cash flow edit errors, warnings, and info, navigate to the ftpshare/logs folder and view the CashFlowEditsProcessExecution.log file.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Cash Flow Edit Process Errors
  The following are the errors, warnings, or information messages displayed by the Cash Flow Edit Process.

17.3.1 Using Cash Flow UI

To execute the cash flow edits process

1. Navigate to the Cash Flow Edits Summary page.
2. Search for a rule.
3. Click in the **Action** column and select **Execute** to execute an existing Cash Flow Edit Rule. The Run Parameter Execution Window is displayed.
4. Select the **As of Date** (Execution Date) and **Legal Entity**, and then click **Run**. The Cash Flow Edits Run confirmation page is displayed. The status of the process is displayed in the Status column. After completion of the process, you can navigate to the Execution Details page by selecting the Execution Details option under the Action column.

> **Note**
>
> You can view the results of running a Cash Flow Edits Rule before the system updates the underlying records in the Instrument tables, provided you selected Preview Mode while defining it. If the rule runs in Preview Mode, query the FSI_CFE_MESSAGES_MLS table for any generated errors.

17.3.2 Batch Based Procedure

Cash Flow Edit Processes can also be executed as Batch Processes. The following approaches are available:

- Simplified Batch
- ICC Batch
- Run Rule Framework
- Simplified Batch
  This section describes the procedure to execute the Cash Flow Edit rule using simplified batch.
- ICC Batch
  This section describes the procedure to execute the Cash Flow Edit rule using ICC Batch framework.
- Run Rule Framework
  This section describes the procedure to execute the Cash Flow Edit rule using the Run Rule Framework.

17.3.2.1 Simplified Batch

This section describes the procedure to execute the Cash Flow Edit rule using simplified batch.

To execute the batch with Simplified Batch

1. From the LHS menu, select **Financial Services Applications**, select **Administration**, and then select **Simplified Batch**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Create a New Simplified Batch.
3. Select **Task Type** as **Transform Data**.
4. Select **Historical Prepayment Batch** from the list of **Seeded Process** types.
5. Input the following in the Optional Parameters text box:

   'Prepaid Detection Process System ID','User ID', for example, '200883','TPUSER'
6. Save the rule and execute.

17.3.2.2 ICC Batch

This section describes the procedure to execute the Cash Flow Edit rule using ICC Batch framework.

To execute the batch with ICC Batch, follow these steps:

1. From the LHS menu, select **Financial Services Applications**, select **Operations**, and then select **Batch Maintenance**.
2. Create a new batch.
3. Enter the Batch Name and Description.
4. To add a task, click the Add button under the Task Details Section.
5. Define the Task ID and Description.
6. Select Components as “WORKFLOW_EXECUTION”.
7. Do not change the default values populated for the following fields:
   - Datastore Type
   - Datastore Name
   - Primary IP for Runtime processes
8. Input the following required parameters:
   - Object ID: $objectId
   - Workflow: “CASH_FLOW_PROCESS_EXEC” or "Cash_FLOW_EDITS_PROCESS_EXEC"
   - Optional Parameters: “RUN_P1_DATE:EXECDATE,RUN_P2_INFODOM:OFSCFEINFO,RUN_P3_ID:200023,RUN_P4_USERID:CFEUSER,RUN_P5_DATE_FORMAT:yyyyMMdd,RUN_P6_AS_OF_DATE:EXECDATE,RUN_P7_LEGAL_ENTITY_ID:-1,RUN_P8_LEGAL_HIERARCHY_ID:-1”.

   The value for parameter “RUN_P3_ID” should be taken from column “PROCESS_ID” of table “FSI_M_CFE_PROCESS_DETAILS” for Cash Flow Process OR column “PROCESS_ID” of table "FSI_M_CFE_EDITS_DETAILS" for Cash Flow Edits. Replace the value 200023 given against “RUN_P3_ID” with the value available in the table for the respective Business Definition. Replace the value CFEUSER given against “RUN_P4_USERID” with the logged-in user id which will be used to audit the last executed by.
   Replace the value -1 given against “RUN_P7_LEGAL_ENTITY_ID” with the actual Legal Entity ID to process the instruments.
9. Save the rule and execute the Batch with Batch ID and MIS Date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

17.3.2.3 Run Rule Framework

This section describes the procedure to execute the Cash Flow Edit rule using the Run Rule Framework.

To execute the batch with Run Rule Framework:

1. From the LHS menu, select **Financial Services Applications**, select **Manage Runs**, and then select **Process**.
2. Create a new rule.
3. Define the Folder, Code, and Name of rule.
4. Select the Component as Transform Data.
5. Select Historical Prepayment Batch from the list of seeded process types.
6. Input the following required parameters:
   - Prepaid Detection Process System ID
   - User Name

   These parameters must be given in the following format (sys Id, User Id), for example, 101000, TPUSER.
7. Save the rule and execute.

17.3.3 Using the Command Line

This topic describes the execution of the Enhanced Cash Flow Edits process for Funds Transfer Pricing (FTP) from the command line in Oracle Financial Services Analytical Applications (OFSAA) Release 8.1. Oracle recommends executing the Enhanced Cash Flow Edits process through the user interface; command-line execution is intended for advanced or administrative use cases.

1. Navigate to `ficdb/bin` folder.
2. Update the below parameters as required and run the command:

```
./CashFlowEditsProcessExecution.sh
<Run_Id>,<PMF_Id>,<Execution_Id>,<Run_Skey>,<Execution_Date>,<INF
ODOM>,<Cash_Flow_Edits_Process_Id>,<User_Id>,<Date_Format>,<As_of
_Date or
MIS_Date>,<Legal_Entity_Id>,<Legal_Entity_Hierarchy_Numeric_Id>
```

   Example:

```
./CashFlowEditsProcessExecution.sh
"$RUNID=1608286329518","$PHID=NA","$EXEID=1761732911543","$RUNSK=
OFS ALM User Guide | 166
22","12/31/2022","OFSALMINFO","202489","105545","MM/dd/yyyy","12/
31/2022","-1","-1"
```

Use the following command to execute the Enhanced Cash Flow Edits process:

```
./CashFlowEditsProcessExecution.sh \
"$RUNID=1608286329518","$PHID=NA","$EXEID=1761732911543","$RUNSK=22",\
"12/31/2022","OFSALMINFO","202489","CFEUSER","MM/dd/
yyyy","12/31/2022","-1","-1"
```

The following table describes the parameters used in the command-line execution:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 17-1 Parameters used in the command-line execution

Parameter Order	Parameter Name	Example Value	Required	Description
1	Run Id	$RUNID=1608286329518	Yes	Any random value can be provided. This value is auto-generated when executed from Batch.
2	PMF Id	$PHID=NA	Yes	Any random value can be provided. This value is auto-generated when executed from Batch.
3	Execution Id	$EXEID=1761732911543	Yes	Any random value can be provided. This value is auto-generated when executed from Batch.
4	Run Skey	$RUNSK=22	Yes	Any random value can be provided. This value is auto-generated when executed from Batch.
5	Execution Date	12/31/2022	Yes	Calendar date or system date of execution.
6	INFODOM	OFSALMINFO	Yes	Infodom in which OFSAA is installed and executed.
7	Cash Flow Edits Process Id	202489	Yes	Numerix identifier of the Cash Flow Edits process.
8	User Id	CFEUSER	Yes	User name of the person executing the process.
9	Date Format	MM/dd/yyyy	Yes	Format in which the MIS date is provided.
10	As of Date / MIS Date	12/31/2022	Yes	As-of date (MIS date) used as a filter to select data from instrument tables.
11	Legal Entity Id	-1	Yes	Numeric ID of the Legal Entity for which instrument data must be processed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 17-1 (Cont.) Parameters used in the command-line execution

Parameter Order	Parameter Name	Example Value	Required	Description
12	`Legal Entity Hierarchy Numeric Id`	`-1`	No	Numeric ID of the Legal Entity hierarchy. This is required when processing a parent Legal Entity and including its child and descendant entities.

17.3.4 View Cash Flow Edit Process Log

To view the cash flow edit errors, warnings, and info, navigate to the ftpshare/logs folder and view the CashFlowEditsProcessExecution.log file.

You can view the log using Batch Monitor also. To view the logs from Batch Monitor, follow these steps:

1. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Monitor**.
2. Select the **Batch Run ID**.
3. View the Task Details of CashFlowEditsExecution.
4. Click **View Log**.

17.3.5 Cash Flow Edit Process Errors

The following are the errors, warnings, or information messages displayed by the Cash Flow Edit Process.

- **Error**: Engine does not process, however sometimes the default value can get used for calculations.
- **Warning**: Engine may use the default value or given wrong data for calculation, results may be incorrect.
- **Info**: Does not impact any processing but results may not be as expected.

Copyright © 2011, 2026, Oracle and/or its affiliates.

18

User Defined Behavior Patterns

User defined behavior patterns allow you to define principal amortization schedules and replicating portfolio characteristics for non-maturity products in your portfolio. You can utilize a behavior pattern while generating cash flows (for use in ALM and Liquidity Risk Management) by entering the behavior pattern code as the amortization type code (AMRT_TYPE_CD) for the relevant instrument records. In Funds Transfer Pricing, for certain TP Methods, you can select a Behavior Pattern to support your Transfer Pricing assumptions. In many cases, particularly for ALM processing, the "non-maturity" instruments will be aggregated or summarized balances.

The Behavior Pattern codes can range from 70000 to 99999

The procedure for working with and managing Behavior Patterns includes the following steps:

- Searching for Behavior Pattern
- Creating a Behavior Pattern
- Viewing and Editing Behavior Patterns
- Copying Behavior Patterns
- Deleting Behavior Patterns
- Creating Behavior Patterns
- Export and Import Replicating Portfolio Data in Excel

18.1 Creating Behavior Patterns

Depending on the Transfer Pricing method, the Behavior Pattern mapped to the individual instrument records (amrt_type_cd), may or may not be used. For cash flow TP methods, the engine will read the Behavior Pattern from the instrument record. For Tractor, Caterpillar and Weighted Average Perpetual methods, the Behavior Pattern is assigned directly within the TP Rule at a Product / Currency level and hence, the TP engine will not refer to the Behavior Pattern assigned to the individual instrument records for these methods.

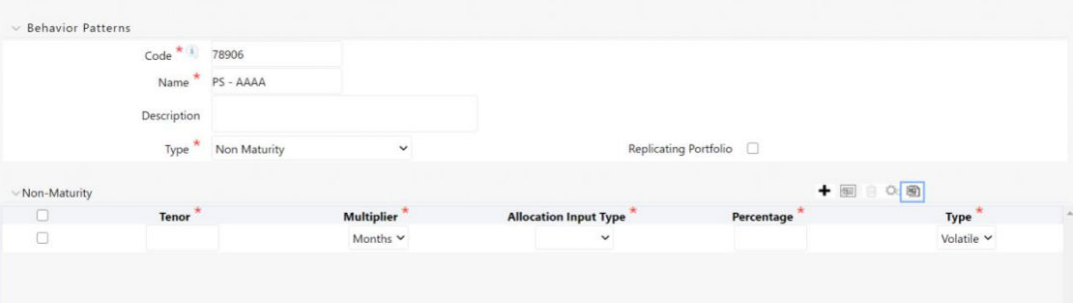
You create Behavior Patterns to capture the principal run-off behavior of product types that do not have contractual maturities.

To create a Behavior Pattern, perform the following:

1. Navigate to the **Behavior Pattern Summary** Page.
2. Click **Add** to display the Behavior Pattern Details Page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 18-1 Behavior Patterns Details Page



3. Enter a unique Numeric Code for the new Behavior Pattern. The code is must be mapped the appropriate instrument record's AMRT_TYPE_CD and BEHAVIOUR_PATTERN_CD to connect the instrument to the appropriate pattern.

> **Note**
>
> The code, also known as an amortization type code, is a numeric identifier for the behavior pattern. The code value must be a number between 70000 and 99999. 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_CD field) to connect the instrument to the appropriate pattern.

4. Enter the **Name** and a **Description** for the pattern.
5. Select the Behavior Pattern Type from the following options:
   - Non Maturity
   - Non-Performing
   - Devolvement and Recovery.

> **Note**
>
> The Replicating Portfolio option is enabled only for the Non Maturity Behavior Pattern. This option should be selected only when defining a Behavior Pattern for use in Funds Transfer Pricing – Tractor TP Method. Replicating Portfolio Behavior Pattern codes should not be mapped to instrument records (amrt_type_cd) and will not be available for selection in any UI other than Tractor TP Method.

6. Define the Behavior Pattern Tenor Specifications for the Maturity Branches.
7. The selection of the Behavior Pattern Type made in the previous step determines the information you must provide to successfully define that Pattern Type. For more information, see:
   - Defining Non-Maturity Behavior Patterns
   - Defining Non-Performing Behavior Patterns
   - Defining Devolvement and Recovery Behavior Patterns

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Behavior Pattern Details Page above displays the specifications associated with the Non Maturity Pattern Type. Should you change this value for one of the other two alternatives, Non Performing or Devolvement and Recovery, the payment specifications section corresponding to the new Pattern Type get refreshed. Although you can change your selection of the Pattern Type at any point in this procedure, sometimes this might result in loss of data related to any prior selection.

- Searching for Behavior Patterns
- Defining Non-Maturity Behavior Patterns (for Tractor TP Method use)
- Defining Non-Performing Behavior Patterns
- Defining Devolvement and Recovery Behavior Patterns
- Defining Non-Maturity Behavior Patterns

18.1.1 Searching for Behavior Patterns

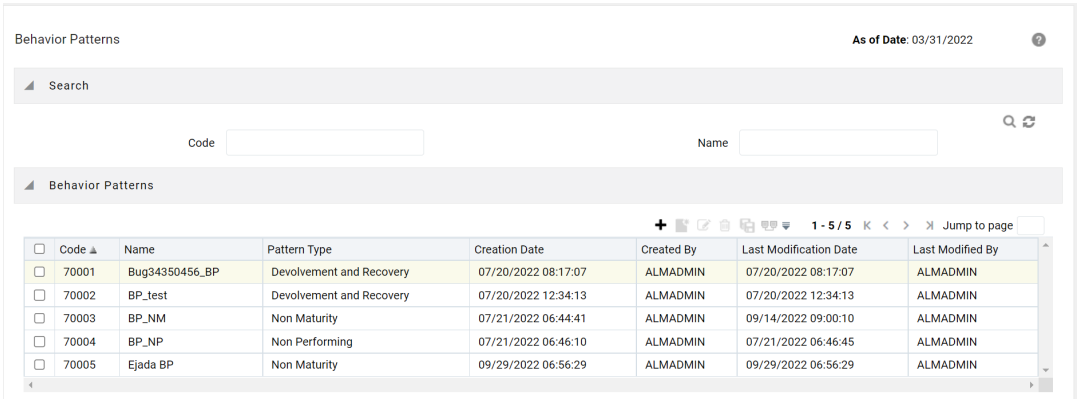
Search for a behavior pattern to perform any of the following tasks:

- View
- Edit
- Copy
- Delete
- Check Dependencies

To search the behavior pattern, follow these steps:

1. Navigate to the **Behavior Pattern Summary** page.

Figure 18-2 Behavior Pattern Summary Page



2. Enter the search criteria.
3. Enter the code or name of the Pattern.
4. Click the **Search** icon.

Copyright © 2011, 2026, Oracle and/or its affiliates.

5. The search results are displayed in a table containing all of the behavior patterns that meet the search criteria. The **Behavior Pattern Summary** page offers several icons that allow you to perform different functions when a behavior pattern is selected.
6. To select a behavior pattern, select a check box in the first column of the table. More than one behavior pattern can be selected at a time, but this disables some of the icons.

You can select or deselect all the behavior patterns in the Summary page by selecting the check box in the upper left-hand corner of the Summary table directly to the left of the Code column header.

18.1.2 Defining Non-Maturity Behavior Patterns (for Tractor TP Method use)

> **Note**
>
> This is applicable only for FTP application.

The Tractor Transfer Pricing Method, utilizes a replicating portfolio concept. Replicating Portfolio's are a special type of Non-Maturity Behavior Pattern and are created and managed through the Non-Maturity Behavior Pattern UI.

Through the replicating portfolio UI, users can define one or more core balance amounts. Users assign a term to each core and generate balance strips at any granularity (e.g. typically daily or monthly, depending on the frequency of the transfer pricing process). To maintain the portfolio over time, users must roll and re-balance the portfolio to update the volatile plug amount and if needed re-balance the core amount.

Update the Balance type when source table is the instrument table. The Balance type allows you to select the type of the Balance.

- If the Source is selected as “Management Ledger, then it can be either Average Balance or Ending Balance.
- If the source is selected as either “Instrument” or: Aggregate Table”, then it can be Cur Book, Cur Par or Average balance.

To define a non-maturity behavior pattern for Tractor TP method use, follow these steps:

1. In the Behavior Pattern details page, select Non Maturity as the Behavior Pattern Type.
2. Enter the following details:

Table 18-1 Key Terms used in the Behavior Pattern Details page

Term	Description
Core Allocation Input	This drop-down is a mandatory selection. The drop-down values are Amount and Percentage. If the Non-Maturity profile method is selected as the Non-Maturity Products Model, then allocation will always be in percentage.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 18-1 (Cont.) Key Terms used in the Behavior Pattern Details page

Term	Description
Core Allocation	The amount that is apportioned to each core. The Allocation amount in absolute or percentage for each strip is automatically determined (when Strip Tenor is "Days", or "Months") by evenly spreading the defined Core Amount across the number of strips for the portfolio. When Strip Tenor is "At Maturity", the entire amount is placed at the maximum term, determined by the inputs for "Term " and "Multiplier". If the user selects 'Amount' in the Core Allocation Input drop-down , then the 'Core Allocation' text box accepts values from 0.0000 to 9,999,999,999.9999. If the user selects 'Percentage' in the Core Allocation Input drop-down , then the 'Core Allocation' text box accepts only a percentage value between 0.0000 and 100.0000.
Tenor	Used to specify the maturity term for the portfolio. This term defines the maximum term for the initial strip balance and the rollover term for each rollover strip. When you manually select or schedule the replicating portfolio process, the Org term of the volatile strip is calculated. Replicating Portfolio generates the volatile strips that sets the ORG_TERM equal to the Period between Maturity date and Origination date.
Multiplier	The unit of time applied to the Tenor. The choices are: • Days • Months • Years
Type	For replicating portfolio, type is defaulted to core. The Volatile strip is generated automatically as a reconciling plug entry to balance the portfolio. The term of the plug entry is defaulted to 1 Day, unless a holiday calendar is used, in which case the volatile amount maturity can be extended to the next business day.
Strip Tenor	Indicates the frequency of the strips generated for the portfolio. Available options are "Days", "Months" or "At Maturity". For example, if "Days" is selected, strips will be generated for each day from the as-of-date to the maturity term. If "At Maturity" is selected, a single strip will be generated and the balance will be placed at the maturity term.
Source Balance Selection	The source balance selection allows you to define the source Instrument Table, Aggregate table (or Ledger Table).
Balance Type	The Balance type allows you to select the type of Balance. It can be either Average Balance or Ending Balance. This option will be enabled if the Source Balance is selected as Management Ledger.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 18-1 (Cont.) Key Terms used in the Behavior Pattern Details page

Term	Description
Enable Holiday Calendar	Replicating Portfolio's allow users to enable a holiday calendar. If this option is selected, portfolio strips will not be generated on weekends or holidays. In addition, during rollover of maturing strips new maturity dates will be adjusted to ensure maturities fall only on working days.
Holiday Calendar Code	The holiday calendar code allows users to select the applicable holiday calendar.
Holiday Calendar Rolling Convention	The rolling convention within replicating portfolios is defaulted to next business day. Related to this method is an additional date adjustment to ensure that only one core strip falls on a single date. We refer to this secondary adjustment as an exclusive business day convention.
Default Portfolio Roll Frequency	The default portfolio roll frequency option allows you to set default rolling frequency of replicating portfolio.
Merge Delta Strips on Re-balancing	If Merge Delta Strips on Re-balancing option is enabled, then the core strips will be merged during the rebalancing.
Generate the Portfolio	After initially creating (and saving) the replicating portfolio definition, users should Generate the Portfolio. This action launches a background process which generates the strip records for the portfolio. Prior to running this process be mindful of the As-of-date defined in your Application Preferences, as this date will be used as the initial Origination Date for the newly created strips. If % is selected as the Core Allocation Input type, the procedure will read the current period balance (CUR_BOOK_BAL or Ending Balance), for the selected “Product Member” (from Source Balance selection) and determine the Required Core Amount based on the resulting Balance x Core %. This applies to both Management Ledger (Ending Balance) and Instrument table (CUR_BOOK_BAL).
Roll the Portfolio Forward	Each period (day or month), users will need to roll the portfolio forward. The new as-of-date for the portfolio will be determined based on the existing as-of-date plus the default roll frequency. As a general rule, users should update their as-of-date in application preferences prior to running the Roll Portfolio Process.
Roll the Portfolio Backwards	This option allows you to roll back the portfolio to the initially selected As-of-Date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 18-1 (Cont.) Key Terms used in the Behavior Pattern Details page

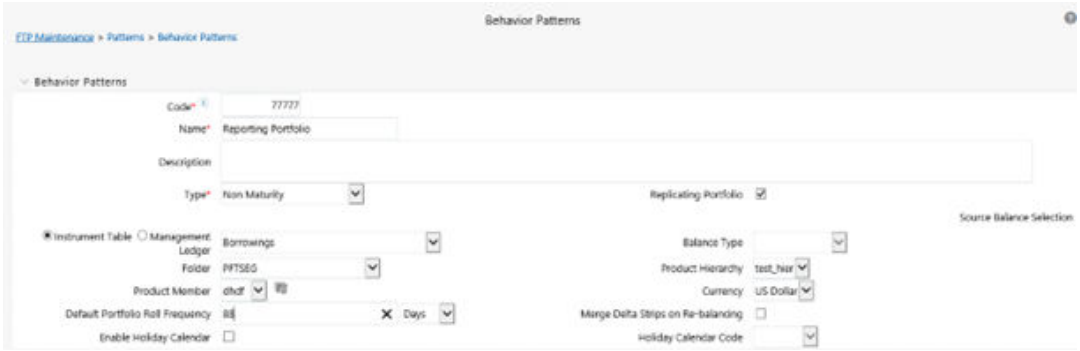
Term	Description
Roll Back	If you have rolled the portfolio forward by mistake and needed to roll the portfolio back to original state, click the Roll Back button. This will first take your confirmation on the rolled back date based on the rolled forward as of date minus the default roll frequency. All the original strips and corresponding tenors will be restored to the original state.
Re-balance the Portfolio	After rolling the portfolio, users will need to re-balance the portfolio. There are two options provided for re-balancing: • **Plug to Volatile Strip**: This option should be selected when no changes to the core allocation have been made. This process will compare the current period source balance with the current portfolio strip balance. The difference will be posted to the new volatile strip. The Plug to Volatile Strip Re-balance method will not be relevant when the Core input type is % as the portfolio balance will change with every new as-of-date and new balancing/delta strips will be required to re-balance the portfolio. • **Rebalance Core Strips**: This option should be selected only when users have modified the Core Allocation or when the Core input type is %. If the Core Allocation has increased or decreased, balancing strips will be generated for each tenor to bring the core strip balance back in line with the Core Allocation Balance. This process will additionally run the Plug to Volatile process to create the plug strip.
View the Portfolio	This option allows you to view the portfolio strips.

This table describes various fields in the Behavior Pattern Details page. You can enter or select the relevant details to populate the screen to define the Non-maturity Behavior Patterns for the Tractor TP Method use.

3. Make your required selections in the **Source Balance Selection** section.
4. Click **Add Core** (one or more) to input the core amount, associated maturity term and strip frequency.
5. To delete a row, select the check box corresponding to the row you want to remove and click **Delete**.
6. Click **Save**.
   The Behavior Pattern is saved and the Behavior Pattern summary page is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 18-3 Behavior Pattern summary page



7. Return to the Behavior Pattern / Replicating Portfolio in EDIT mode and execute the portfolio maintenance activities, including:
   - Generate Portfolio
   - Roll Portfolio
   - Rebalance Portfolio
   - View Portfolio

> **Note**
>
> After the Replicating Portfolio is generated and the volatile plug has been updated for the current period, it is ready for processing by the FTP Engine. FTP Processes utilizing the Tractor TP Method should not be Run until all Replicating portfolios are updated.

18.1.3 Defining Non-Performing Behavior Patterns

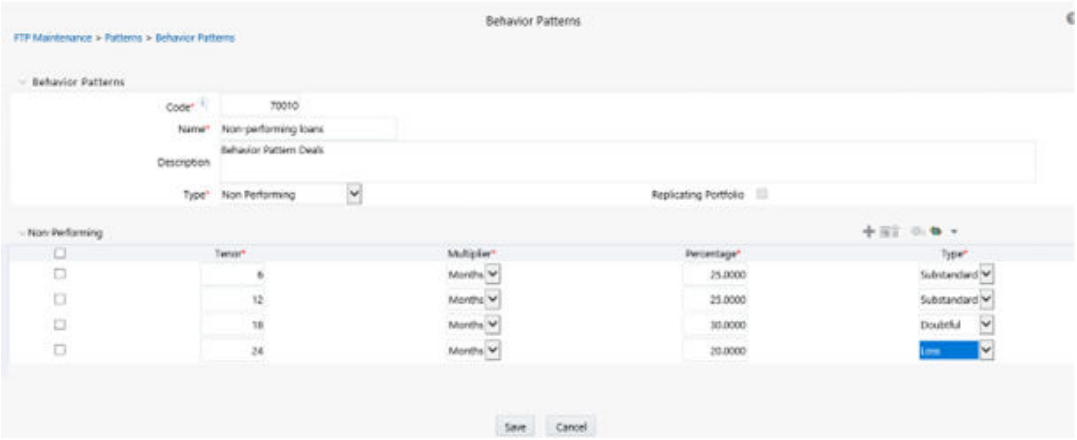
Non-Performing Behavior Patterns are commonly used for balances that are classified as non-earning assets. These balances are typically sourced from the Management Ledger as aggregate balances. Users can assign expected maturity profiles to these balances classifying them into appropriate categories of Sub Standard, Doubtful, or Loss.

To define the Non-Performing Behavior Patterns, perform the following steps:

1. In the Behavior Pattern Details Page, select **Non-Performing** as the **Behavior Pattern Type**.
2. Enter the **Code**, **Name**, and **Description** for the Behavior Pattern.
3. Click the **Add** icon to open the **Non-Performing Behavior Patterns Summary** page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 18-4 Behavior Pattern with Type as Non-Performing



4. Enter or select the following details:
   - **Tenor**: Specify the maturity tenor for the first maturity strip. For example, if “1 Day” is defined, then the applicable percentage of the balance will runoff (mature) on the As-of-Date + 1 Day.
   - **Multiplier**: The unit of time applied to the Tenor. The choices are:
     - Days
     - Months
     - Years
   - **Percentage**: The relative amount of the Principal Balance that will mature on the date specified by the Tenor + Multiplier. The percentage amounts can exceed 100% for Non-Performing Patterns.
   - **Runoff Type**: This allows you to classify the Runoff based on the appropriate type. The options are:
     - Substandard
     - Doubtful
     - Loss

   > **Note**
   >
   > There is no difference in behavior from a Cash Flow perspective, but the Runoff Amount will be written to a Principal Runoff Financial Element corresponding to the selected Runoff Type.

5. Click the **Add** icon to add additional payment strips to the Pattern and define appropriate assumptions for each strip.
6. To delete a row, select the check box corresponding to the row(s) you want to remove and click the **Delete** icon.
7. Click **Save**.

The Behavior Pattern is saved and the **Behavior Pattern Summary** Page is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

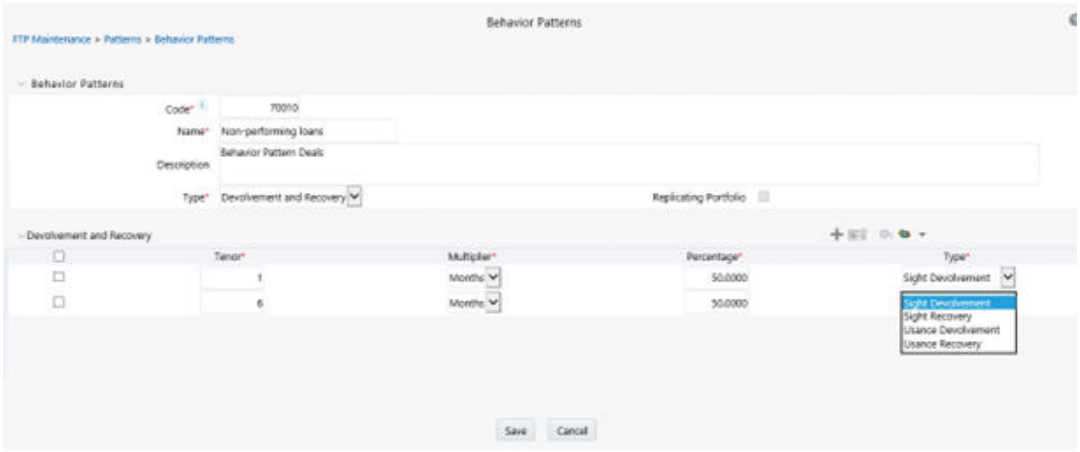
18.1.4 Defining Devolvement and Recovery Behavior Patterns

Devolvement and Recovery Behavior Patterns are commonly used for estimating Cash Flows associated with Letters of Credit and Guarantees. These product types are categorized as Off-Balance-Sheet Accounts. Users can assign expected maturity profiles to the related balances classifying them into appropriate categories of Sight Devolvement and Sight Recovery or Usance Devolvement and Usance Recovery. Sight Devolvement and Recovery are the most common types.

To define the Non-Performing Behavior Patterns, perform the following steps:

1. In the **Behavior Pattern Details** page, select **Devolvement and Recovery** as the Behavior Pattern Type.
2. Enter the **Code**, **Name**, and **Description** for the Behavior Pattern.
3. Click the **Add** icon to open the Non-Performing Behavior Patterns Summary page.

Figure 18-5 Behavior Pattern with Type as Devolvement and Recovery



4. Enter or select the following details:
   - **Tenor**: Specify the maturity tenor for the first maturity strip. For example, if “1 Day” is defined, then the applicable percentage of the balance will Runoff (mature) on the As-of-Date + 1 Day.
   - **Multiplier**: The unit of time applied to the Tenor. The choices are:
     - Days
     - Months
     - Years
   - **Percentage**: The relative amount of the Principal Balance that will mature on the date specified by the Tenor + Multiplier. The percentage amounts can exceed 100% for devolvement and recovery patterns.
   - **Runoff Type**: This allows you to classify the Runoff based on the appropriate type. The options are:
     - **Sight Devolvement**: indicates the Beneficiary is paid as soon as the Paying Bank has determined that all necessary documents are in order. This is the preferred approach.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Sight Recovery**
- **Usance Devolvement**: Usance is a period, which can be between 30 and 180 days after the bill of Lading Date.
- **Usance Recovery**

> **Note**
>
> There is no difference in behavior from a Cash Flow perspective, but the Runoff Amount will be written to a Principal Runoff Financial Element corresponding to the selected Runoff Type.

5. Click the **Add** icon to add additional payment strips to the Pattern and define appropriate assumptions for each strip.
6. To delete a row, select the check box corresponding to the row(s) you want to remove and click the **Delete** icon.
7. Click **Save**.
   The Behavior Pattern is saved and the Behavior Pattern Summary Page is displayed.

This release includes a facility that enables the automatic load of behavior patterns. There is a Behavior Pattern Loader Utility, which helps to automatically load model output records from STG_BEHAVIOUR_PATTERN into FSI_BEHAVIOUR_PATTERN_MASTER and FSI_BEHAVIOUR_PATTERN_DETAIL. The utility first performs a sanity check on all records as per business requirement and only moves records that pass these checks. In the case of Non-Maturing Behavior Patterns, only Non-Replicating Portfolio Patterns can be loaded through this utility. For more information, see the OFS Data Model Utilities User Guide.

18.1.5 Defining Non-Maturity Behavior Patterns

Non-Maturity Behavior Patterns are commonly used for deposit products like Checking, Savings, and Money Market Accounts as well as for Credit Card Accounts. These account types are similar in that they do not have Contractual Cash Flows because Customers have the option to deposit or withdraw any amount at any time (up to any established limits).

When working with Non-Maturity Behavior Patterns, your percentage weights, assigned to maturity terms must add up to 100%.

To define a non-maturity behavior pattern for non-Tractor TP method, there are two ways:

- Manual
- Non-Maturity Products Model

Figure 18-6 Behavior Patterns



Copyright © 2011, 2026, Oracle and/or its affiliates.

For Manual Model, you can follow these steps:

1. In the Behavior Pattern Details page, select Non Maturity as the Behavior Pattern Type.
2. Select Non-Maturity Products Profile Method as **Manual**.
3. Enter or select the following details:
   - **Tenor**: Used to specify the maturity term for the particular row. For example, if “1 Day” is defined, then the applicable percentage of the balance will runoff (mature) on the As-of-Date + 1 Day.
   - **Multiplier**: The unit of time applied to the tenor. The choices are as follows:
     - Days
     - Months
     - Years
   - **Percentage**: The outstanding balance indicating how much of the outstanding balance will mature on the specified term. If the Non-Maturity profile method is selected as the Non-Maturity Products Model, then allocation will be in percentage.
   - **Allocation Input Type**: This field allows you to select the Amount or Percentage when defining the volume for each maturity tier.
   - **Type**: Allows you to classify the runoff based on the appropriate type. If you select Percentage under 'Allocation Input Type', this allows you to select Core or Volatile. When Amount 'Allocation Input Type' is selected, only Core amounts are allowed to be entered. The TP engine calculates the Volatile amount internally based on the sum of the portfolio balance less the sum of the core amounts. i.e. 100% of the portfolio balance is accounted for as either core or volatile.

   > **Note**
   >
   > There is no difference in behavior from a cash flow perspective, but the runoff amount will be written to a Principal Runoff financial element corresponding to the selected Runoff Type.

4. Click the **Add** icon to add additional payment strips to the Pattern. After defining the initial strip as Volatile, subsequent strips are usually classified as Core with varying maturity terms assigned.
5. You can also use the Excel Import/Export feature to add the Behavior Pattern information. For more information, see Excel Import/Export Feature.
6. You may want to utilize the Data Input Helper to copy from a row where you have already defined the Pattern definition or apply a fixed value down the page. For more information, see the Data Input Helper feature.
7. To delete a row, select the check box corresponding to the row you want to remove and click the **Delete** icon.
8. Click **Save**.
   The Behavior Pattern is saved and the Behavior Pattern Summary Page is displayed.

Another way to define Non-Maturity profile is via Non-Maturity Products Model. You can have single Core or Multiple Core strips based on if decay profile is being used:

- One Core and One Volatile strip in percentage if “Multiple Core Strips as per Decay Profile” is not selected.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Multiple Core and Volatile strips in percentage if “Multiple Core Strips as per Decay Profile” is selected.

Single Core Non-Maturity Behavior Pattern: Behavior pattern picks the core from the underlying model which is defined at certain portfolio level. the application calculates the percentage of the core as (Core/total)*100, and the remaining would become volatile strip with tenor 1 day. Core strip tenor will be WAL given by the model. The amount is not applicable if the model is being used.

Multiple Core Non-Maturity Behavior Pattern: The Multiple Core Strips as per Decay Profile” check-box allows you to define Multiple Core Strips.

> **Note**
>
> Users have an option to define Tenor and multiplier for each strip for the concerned Behavior Pattern. System will calculate the strip percentage as per the selected model's decay profile and user -defined strip tenor.

Total portfolio balance by Model (Model Output): Total balance is the balance available at the latest As of Date in the Historical Period selected for which model is created. For example, if the Historical Period is 1st Jan 2020 to 31st Dec 2020, and say most recent as of date was 31st Dec 2020, then whatever is the total available balance of all the accounts considered under portfolio (product, currency,); would represent the total Portfolio Balance.

Threshold balance: This field is displayed only when the “Multiple Core Strips as per Decay Profile” check-box is selected. The default value of this field is 0.0. This field accepts only positive numbers/Decimals values. While using decay rate profile from the underling model, balance will keep reducing using every period’s decay rate till threshold balance is reached . No further strips will be created beyond Threshold balance. It is recommended to give some valid positive value as threshold balance, else after few initial strips, principal will become small and decaying part will be even smaller, so you will get huge number of strips with negligible run-offs and after 500 strips, system will throw an error.

The **Generate Profile** button creates the Profile Strips as per total and threshold balance.

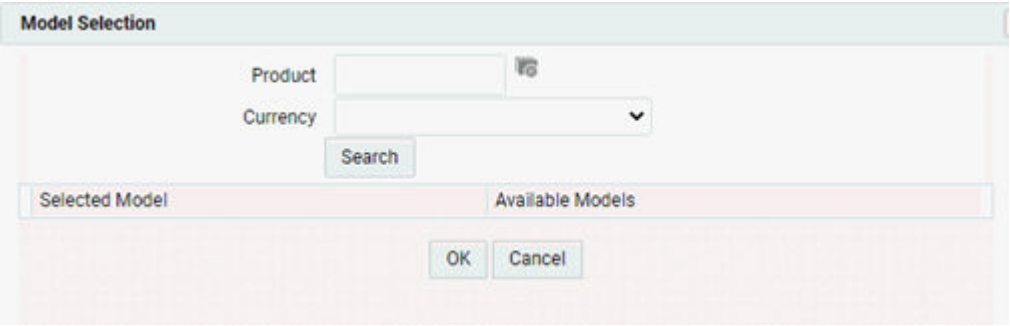
The system uses the decay profile/decay rate generated by the model and creates the remaining balance/Runoff Profile as per As of Date (frequency) which can be in Days/Months and Years. If the frequency is in Days, then the tenor of the defined strips would be days else it would be in months/years as per the model definition.

For example, starting 100,000 total balance, in 1 month, 20,000 will Runoff with 20% decay rate and that becomes the first strip. While using the model only Percentage Strip Mode is allowed, So first strip is defined with 1 Month tenor and thereafter 2nd, 3rd strip with corresponding balance run-offs. And say cut off given by the user was 20,000 so up to 8th month strips would be defined using regular Runoffs as per decay profile. After that whatever is remaining, 16,777.22, in this case, would be given next term point 9 month as that is available balance up to 9 months. You can choose any strip as volatile or core. But in any case total strips percentage cannot be more than 100 By default, the minimum tenor strip will become volatile and all higher tenors will be tagged as Core Strips but you can still edit them.

Use the Model Search option to search the model based on Product and Currency.

Copyright © 2011, 2026, Oracle and/or its affiliates.

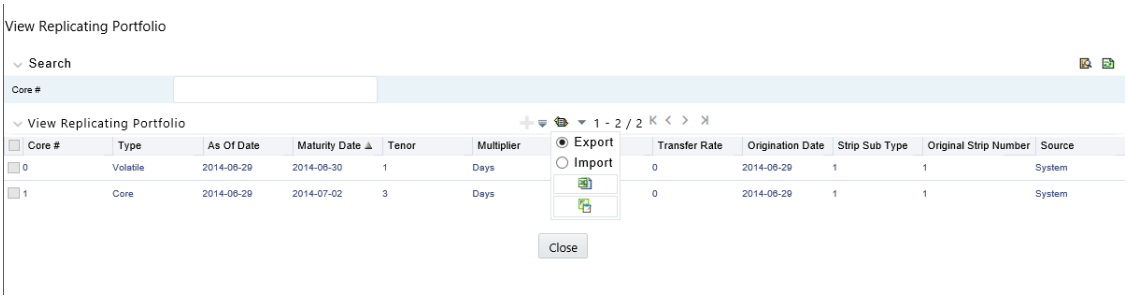
Figure 18-7 Model Selection



18.2 Export and Import Replicating Portfolio Data in Excel

There is an option through the Replicating Portfolio > View Portfolio UI to manually edit existing portfolio strips through Export and Import of the active Strip Data.

Figure 18-8 Export and Import Replicating Portfolio Data in Excel

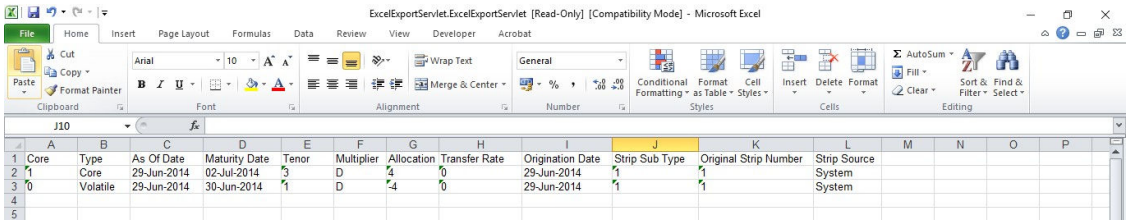


The following screen shot illustrates the functionality:

- The Export option works against the entire active portfolio. For example, a user can currently filter on a specific CORE # or look at results for all Core's. Additionally, the selection of Strips can span multiple pages, for example - as seen in the preceding screen shot, 1-20 of 71. In this example, all 71 active strips are exported to Excel.
- The import function will replace ALL existing “Active” strips.
- The strip data being imported is validated to confirm that all required data is included. If the data is not complete, for example, it does not provide information for Core #, Strip Type, As of Date, Maturity Date, Tenor, Multiplier, Allocation (or Amount), then a warning message is given indicating that “The selected data is incomplete and cannot be imported. Please re-check the data and try again.” The portfolio can also be edited directly on the view portfolio screen after new strips have been imported.
- When you click Strip Source option, the status column in the summary table shows the tagged strip records that have been created by system or manually. You can edit these tags for strips that are manually created or existing strips after exporting them into Excel.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 18-9 Excel Sheet



Copyright © 2011, 2026, Oracle and/or its affiliates.

19

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 by entering the payment pattern code as the amortization type code for the instrument. The Payment Pattern code can range from 1000 to 69999.

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

The procedure for working with and managing Payment Patterns is, similar to that of other Oracle assumption rules.

Topics

- Searching for Patterns
- Creating Payment Patterns
- Viewing and Editing Payment Patterns
- Copying Payment Patterns
- Deleting Payment Patterns
- Search for Payment Patterns
- Create Payment Patterns

19.1 Search for Payment Patterns

Search for a payment pattern to perform any of the following tasks:

- View
- Edit
- Copy
- Delete
- Check Dependencies

Prerequisites

- Predefined Payment Patterns

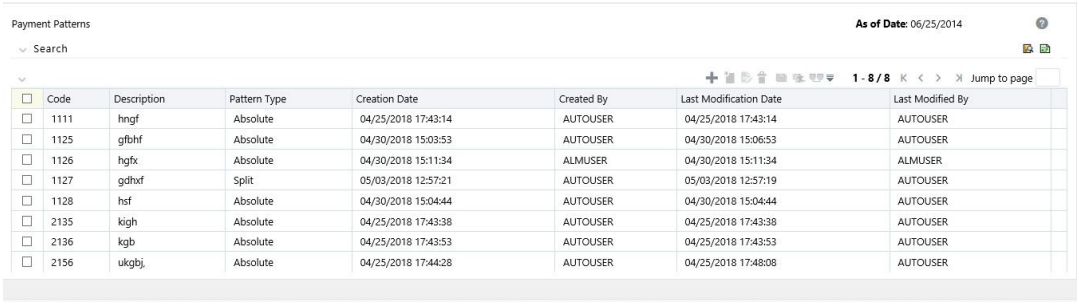
Procedure

To search for payment patterns, do the following:

1. Navigate to the **Payment Pattern** summary 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 19-1 Payment Pattern Summary Page



Code	Description	Pattern Type	Creation Date	Created By	Last Modification Date	Last Modified By
1111	hngf	Absolute	04/25/2018 17:43:14	AUTOUSER	04/25/2018 17:43:14	AUTOUSER
1125	gfbhf	Absolute	04/30/2018 15:03:53	AUTOUSER	04/30/2018 15:06:53	AUTOUSER
1126	hgfx	Absolute	04/30/2018 15:11:34	ALMUSER	04/30/2018 15:11:34	ALMUSER
1127	gdhxf	Split	05/03/2018 12:57:21	AUTOUSER	05/03/2018 12:57:19	AUTOUSER
1128	hsf	Absolute	04/30/2018 15:04:44	AUTOUSER	04/30/2018 15:04:44	AUTOUSER
2135	kigh	Absolute	04/25/2018 17:43:38	AUTOUSER	04/25/2018 17:43:38	AUTOUSER
2136	kgb	Absolute	04/25/2018 17:43:53	AUTOUSER	04/25/2018 17:43:53	AUTOUSER
2156	ukgbj,	Absolute	04/25/2018 17:44:28	AUTOUSER	04/25/2018 17:48:08	AUTOUSER

2. Enter the Code or Description of the pattern in Search Criteria.
3. Click the **Search** icon.

Only patterns that match the search criteria are displayed.

You can control the number of rows to display on-screen by selecting the "Pagination Options" icon from the action bar.

19.2 Create Payment Patterns

You create payment patterns to capture the repayment behavior of instruments that are too complex to be accommodated through the use of the standard instrument table fields.

To create the payment pattern, do the following:

1. Navigate to the **Payment Pattern** summary page.
2. Click **Add** icon. The **Add Payment Pattern** page is displayed.
3. Enter a Code value for the new payment pattern. You can also click Generate Code Option in Code field to generate the code automatically.

> **Note**
>
> The code, also click Generate Code Option in Code field to generate the code automatically 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 69999. 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_CD field) to connect the instrument to the appropriate pattern.

4. Enter the Name for pattern.
5. Enter a brief Description for the pattern.
6. Select the Payment Pattern Type: Absolute, Relative, or Split.
7. Select the Payment Type: Conventional, Level Principal, or Non-amortizing. Define the Payment Pattern Term Specifications for payment phases.

> **Note**
>
> The Payment Type option is not available for Split Payment Pattern type.

Copyright © 2011, 2026, Oracle and/or its affiliates.

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
- Defining Relative Payment Patterns
- Defining Split Payment Patterns

8. Click **Save**.

> **Note**
>
> The Payment Pattern Details 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 loss of data related to any prior selection.

- Define Absolute Payment Patterns
- Define Relative Payment Patterns
- Define Split Payment Patterns

19.2.1 Define Absolute Payment Patterns

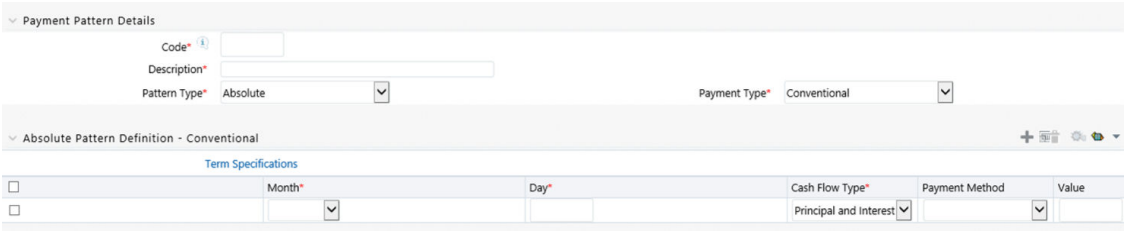
Absolute payment patterns are commonly used for instruments that are on a seasonal schedule, such as agricultural or construction loans that require special payment handling based on months or seasons.

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

Selecting Absolute as the pattern type.

Figure 19-2 Absolute Payment Patterns



To define absolute payment pattern, do the following:

1. In the **Payment Patterns** page, select **Pattern Type** as **Absolute**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Select the Payment Type from the drop-down list: **Conventional**, **Level Principal**, or **Non-Amortizing**. The Payment Type determines the type of information required to successfully define the Payment Phase.
3. Define the **Payment Phases**. A Payment Phase is a set of payment characteristics that defines the timeline of the instrument's amortization.
   a. Define the following parameters:
      - **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.
   b. Select the **Cash Flow Type**. The available types depends on the Payment Type. This do not apply to the Non-Amortizing Payment Type.
   Table: Relationship between Cash Flow Type and Payment Types

	Level Principal	Non-Amortizing	Conventional
Principal and Interest	Yes		Yes
Principal Only	Yes		
Interest Only	Yes	Yes	

   c. Select the **Payment Method**. The available Payment Methods depend on the Payment Type. For more information, see: Relation between Payment Method and Payment Types. Payment Methods do not apply to the Non-Amortizing Payment Type.
   d. Enter the Value for the Payment Method you selected in the previous step for applicable Payment Types.
   If you selected the Interest Only Payment Method in the previous step, the Value field does not apply.
4. Click the **Add** icon to add additional Payment Phases to the Pattern. Click Delete icon corresponding to the rows you want to delete.
5. Click **Add Multiple Row** icon to enter the number of rows you want to add and click Go.
6. The **Download Excel** icon helps you to export payment information, modify and paste back in the UI.

> **Note**
>
> 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.

7. Click **Apply** and Save.

The Payment Pattern is saved and the Payment Pattern summary page is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> When a detail instrument using an Absolute Payment Pattern is processed for Remaining Term 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 Instrument table is always the date used for processing the initial payment.

The following table describes the relationship between Payment Phase properties and Payment Types.

Table 19-1 Relationship between Payment Phase Properties and Payment Types

	Level Principal	Non-Amortizing	Conventional
Month	Yes	Yes	Yes
Day	Yes	Yes	Yes
Payment Method	Yes		Yes
Value	Yes		Yes

The following table describes the relationship between Payment Method and Payment Types.

Table 19-2 Relationship between Payment Methods and Payment Types

Payment Method	Level Principal	Non-Amortizing	Conventional
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

19.2.2 Define Relative Payment Patterns

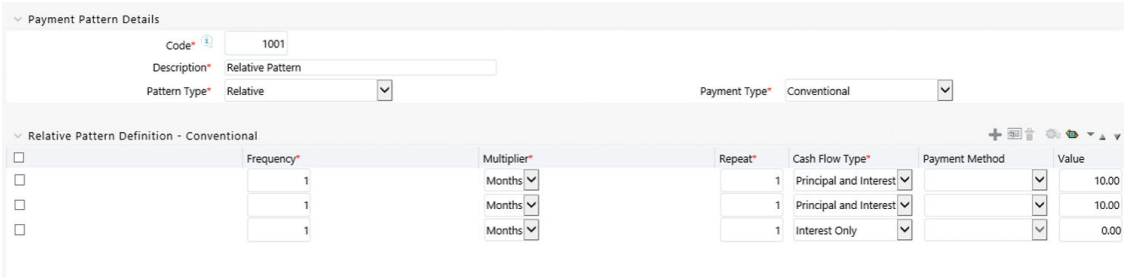
You create Relative Payment patterns for instruments that have irregular scheduled payments.

Prerequisites

Selecting Relative as the pattern type.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 19-3 Relative Payment Patterns



To define a relative payment pattern, follow these steps:

1. In the **Payment Patterns** page, select **Pattern Type** as **Relative**.
2. Select the Payment Type from the drop-down list: **Conventional**, **Level Principal**, or **Non-Amortizing**. The Payment Type determines the type of information required to successfully define the Payment Phase.
   The payment type determines the available characteristics for defining the payment amount.
3. Define the **Payment Phase**.
   The payment type determines the type of information required to successfully define the payment phase. For more details, see: Relation between Payment Phase Attributes and Payment Types.
   a. Enter the **Frequency** for each payment phase.
   b. Select the appropriate Multiplier for each payment phase from the following options:
      - Days
      - Months
      - Years
   c. Enter the number of times each Payment Phase should be repeated in the Repeat column.
   d. Select the **Cash Flow Type**. The available types depend on the Payment Type. This do not apply to the Non-Amortizing Payment Type.

Table 19-3 Relationship between cash Flow Type and Payment Types

	Level Principal	Non-Amortizing	Conventional
Principal and Interest	Yes		Yes
Principal Only	Yes		
Interest Only	Yes	Yes	

   e. Select the **Payment Method**.
   The available payment methods depend on the payment type. For more details, see Relation between Payment Method and Payment Types. Payment Methods do not apply to the Non-Amortizing Payment Type.
   f. Type the Value for the Payment Method you selected in the previous step for applicable Payment Types.
4. Click the **Add** icon to add additional Payment Phases to the Pattern. Click Delete icon corresponding to the rows you want to delete.

Copyright © 2011, 2026, Oracle and/or its affiliates.

5. Click **Add Multiple Row** icon to enter the number of rows you want to add and click Go.
6. The **Download Excel** icon helps you to export payment information, modify and paste back in the UI.

> **Note**
>
> 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.

7. Click **Apply** and **Save**.
   The payment pattern is saved and the Payment Pattern home page is displayed.

   You can use the Move Up and Move Down icons to move the Payment Phases up or down. Also, you can click the Delete icon to delete a row. Any empty rows are ignored and not saved with the payment pattern

> **Note**
>
> It is not necessary to set up relative payment patterns for the complete term of an instrument. The payment pattern automatically repeats until the 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 repeat value 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 in one month. The second row shows 999 payments in 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.

Table 19-4 Relationship between Payment Phases and Payment Types

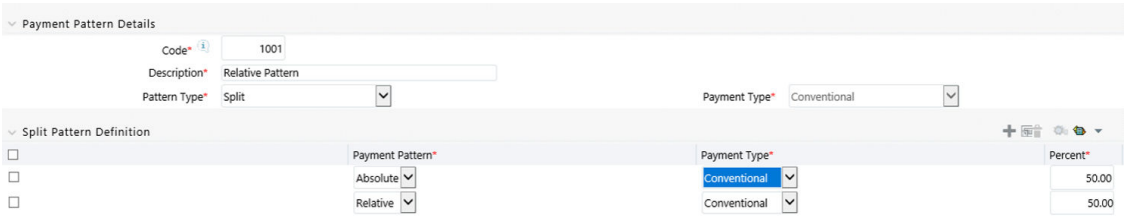
Payment Phase Attributes	Payment Types: Level Principal	Payment Types: Non-Amortizing	Payment Types: Conventional
Frequency	Yes	Yes	Yes
Multiplier	Yes	Yes	Yes
Repeat	Yes	Yes	Yes
Payment Method	Yes		Yes
Value	Yes		Yes

Copyright © 2011, 2026, Oracle and/or its affiliates.

19.2.3 Define Split Payment Patterns

You can use a Split payment pattern for financial instruments that make principal payments along with 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.

Figure 19-4 Split Payment Patterns



To define a split payment pattern, follow these steps:

1. In the Payment Patterns page, select Pattern Type as Split.
2. Define Split Pattern definition.
   a. Select the required **Pattern Sub Type** for each leg.
      - Absolute
      - Relative
   b. Select the Payment Type for each Payment Phase or Split.
   c. Enter the Percent value to indicate the percentage weight of the timeline being defined for the individual payment phases (each row). The sum of the percentage weights must total 100%.

> **Note**
>
> 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:
>
> - Define Absolute Payment Patterns
> - Define Relative Payment Patterns

3. Select one of the legs and then select **Define** button to enter pattern details for the leg.
4. Select one of the legs and then select **Delete** button to delete pattern details for the leg.
5. Click the **Add** icon to add additional Payment Phases to the Pattern.
6. Click **Add Multiple Row** icon to enter the number of rows you want to add and click Go.
7. Click **Apply** and **Save**.

The Split payment pattern is saved and the Payment Pattern summary page is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Copyright © 2011, 2026, Oracle and/or its affiliates.

20

User Defined 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. You can utilize a Repricing Pattern to generating Cash Flows by entering the Adjustable Type Code as “Repricing Pattern” along with actual Repricing Pattern Code for the relevant Instrument Records.

The procedure for working with and managing Repricing Patterns is, similar to that of other Oracle Business Rules.

Topics

- Search Repricing Patterns
- Create a Repricing Pattern
- View and Edit Repricing Patterns
- Copy Repricing Patterns
- Delete Repricing Patterns
- Search Repricing Pattern
- Create Repricing Patterns

20.1 Search Repricing Pattern

Search for a Repricing Pattern to perform any of the following tasks:

- View
- Edit
- Copy
- Delete

Prerequisites

Predefined Repricing Patterns

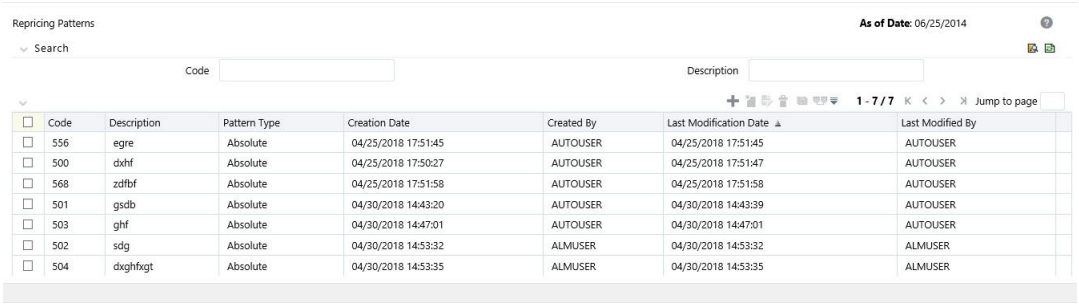
Procedure

To search the repricing patterns, follow these steps:

- Navigate to the **Repricing Pattern Summary** 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 20-1 Repricing Pattern Summary Page



Code	Description	Pattern Type	Creation Date	Created By	Last Modification Date	Last Modified By
556	egre	Absolute	04/25/2018 17:51:45	AUTOUSER	04/25/2018 17:51:45	AUTOUSER
500	dxhf	Absolute	04/25/2018 17:50:27	AUTOUSER	04/25/2018 17:51:47	AUTOUSER
568	zdfbf	Absolute	04/25/2018 17:51:58	AUTOUSER	04/25/2018 17:51:58	AUTOUSER
501	gsdb	Absolute	04/30/2018 14:43:20	AUTOUSER	04/30/2018 14:43:39	AUTOUSER
503	ghf	Absolute	04/30/2018 14:47:01	AUTOUSER	04/30/2018 14:47:01	AUTOUSER
502	sdg	Absolute	04/30/2018 14:53:32	ALMUSER	04/30/2018 14:53:32	ALMUSER
504	dxghfxgt	Absolute	04/30/2018 14:53:35	ALMUSER	04/30/2018 14:53:35	ALMUSER

- Enter the Code or Description of the pattern in Search Criteria.
- Click the **Search** .
  Only patterns that match the search criteria are displayed.

20.2 Create Repricing Patterns

You create Repricing patterns to capture the Repricing Behavior of instruments whose rates change according to complex schedules.

To create the Repricing Pattern, follow these steps:

1. Navigate to Repricing Pattern Summary Page.
2. Click **Add** icon.
   The Add Repricing Pattern Page is displayed.
3. Enter a Numeric Code Value for the new Repricing Pattern. You can also click Generate Code Option in Code field to generate the code automatically.
4. Enter the name of the pattern.

> **Note**
>
> The code is a numeric internal identifier for the repricing pattern. The code value must be a number between 500 and 4999 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_CD and REPRICE_PATTERN_CD (applicable to CFE only)) to connect the instrument to the appropriate pattern.

5. Type a brief description for the pattern.
6. 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
   - Defining Relative Repricing Patterns

   The Add 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

Copyright © 2011, 2026, Oracle and/or its affiliates.

displayed when you change the pattern type. The data is discarded only after your confirmation.

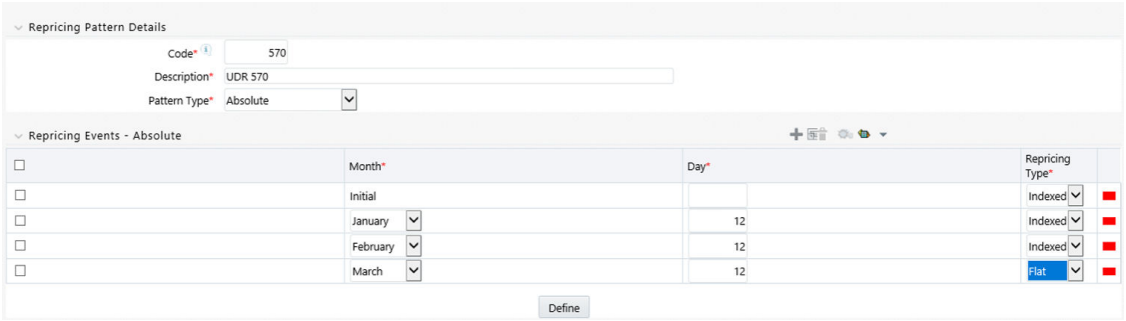
7. Click **Save**.

- Define Absolute Repricing Patterns
- Define Relative Repricing Patterns

20.2.1 Define 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.

Figure 20-2 Define Absolute Repricing Pattern



Prerequisites

Selecting Absolute as the pattern type.

Procedure

This table describes key terms used for this procedure.

Table 20-1 Key Terms used in Absolute Repricing Pattern

Key Terms	Description
Add Row	Allows you to add one or more Repricing Events.
Add Multiple Rows	Allows you to add more Repricing Events.
Define	Add detail for each Repricing Event.
Delete	Allows you to delete specific rows in the Repricing Events Table.
Month	In conjunction with the Day field, this drop-down menu, allows 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.

To define Absolute Repricing Pattern, perform the following:

1. Select Pattern Type as **Absolute**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. This screen allows you to do the following:
   - Add Row: Allows you to Add one or more repricing events.
   - Delete: Allows you to delete specific rows in the Repricing Events table.
3. Click Add Event.
4. Enter the following details:
   - Month: In conjunction with the Day field, this drop-down menu, allows 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.
5. Specify the required month-day combination for the event. You cannot specify a month-day combination for the first event as this row is reserved for the initial period.
6. Select the Repricing Type: **Flat** or **Indexed**.
   The default is flat. If you select Indexed, the system automatically changes the fields available for entry.

   Note: You can change your selection of the Repricing Type at any point in this process. Sometimes it may cause a loss of data.

   For more information on Flat Repricing Type, see . Repricing Event is Flat Repricing

   For more information on Index Repricing Type, see Repricing Event is Index Repricing.
   - Select Balance Tier option:
     - **None**: If selected, the Balance Tiered Pricing is not applied.
     - **Current Balance**: Users can define balance tiers and associate different rates with the corresponding balance tier level. Balance tier in this case is decided using the principal balance of instrument on As of Date. Thus, even when actual balance goes down due to repayment, the instrument continues to be in original balance tier.
     - **Reducing Balance:** If the Balance Type is selected as Reducing Balance, then the repricing rate will be calculated using principal balance as on Repricing Date. Thus, the balance tier applicable to instrument range over its life due to repayment.
7. Click **Define**.

Repricing Event is Flat Repricing

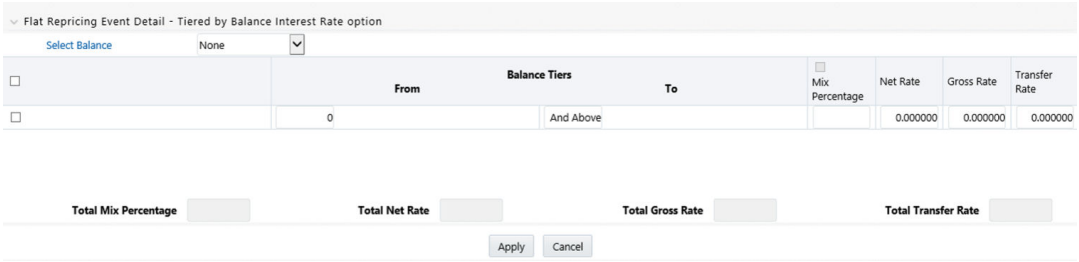
Flat Rate: A Flat Rate is a specific rate—it is directly input.

To define a Flat Rate Event, follow these steps:

1. Select the Flat option from Repricing Method drop-down list for the event you are going to define. Notice the bottom half of the screen refreshes, displaying the required inputs. Complete the following steps on the Add Repricing Events Page:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 20-3 Flat Repricing Pattern Detail Pane



2. Select Balance Tier option:
   None: If selected, the Balance Tiered pricing is not applied.

   Current Balance: Users can define balance tiers and associate different rates with the corresponding balance tier level.
3. Specify the required month-day combination for the event. You cannot specify a month-day combination for the first event as this row is reserved for the initial period.
4. Enter the Net Rate.
5. Enter the Gross Rate.
6. Enter the Transfer Rate.

> **Note**
>
> The Transfer Rate functionality will be released in future.
>
> You must enter a valid value for at least one of these rate fields.

7. Click Apply.
   The Event Summary Page is displayed. Note the status indicator has changed from Red to Green indicating that details for the event have been defined. At this point, you have the option of defining additional events or saving. To add an additional event, repeat Click Add Row. You can edit the details of Defined Event. If you want to save the repricing pattern and events, advance to the next step.

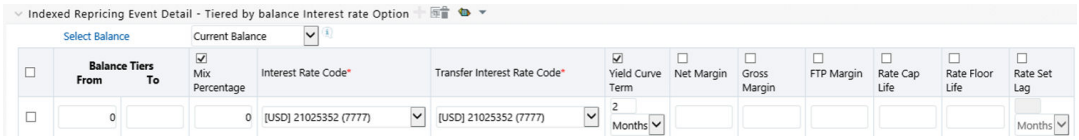
Repricing Event is Indexed Repricing

An Indexed Rate is a set of parameters used to calculate a rate.

To define an Indexed Rate Event, follow these steps:

1. Select the Indexed option from Repricing Method drop-down list for the event you are going to define. Notice the bottom half of the screen refreshes, displaying the required inputs. Complete the following steps on the Add Repricing Events Page:

Figure 20-4 Indexed Repricing Event Detail



2. Select the check box above each column that you want to include in the repricing event.
3. Select a Balance Tier option:
   § None: The Balance Tiered pricing is not applied.

Copyright © 2011, 2026, Oracle and/or its affiliates.

§ Current Balance: The users can define balance tiers and associate different rates with the corresponding balance tier level.

4. Select the Interest Rate Code.
5. Select the Transfer Price Code.

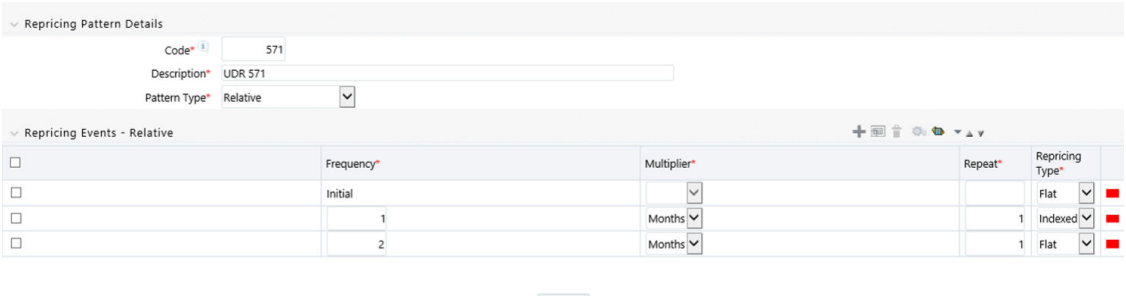
> **Note**
>
> The Transfer Price Curve functionality will be released in the future.

6. Enter the Yield Curve Term and select the appropriate Multiplier.
7. Enter the Net Margin.
8. Enter the Gross Margin.
9. Enter the Transfer Price Margin.
10. Enter the Rate Cap Life.
11. Enter the Rate Floor Life.
12. Enter the Rate Set Lag and select the appropriate Multiplier.
13. Click Apply. The Event Summary Page is displayed.
14. At this point, you have the option of defining additional events or saving. To add an additional event, repeat Click Add Row.
    If you want to save the repricing pattern and events, advance to the next step.
15. Click **Save**.
    The Repricing Pattern is saved and the Repricing Pattern Summary Page is displayed.

20.2.2 Define Relative Repricing Patterns

The Relative Repricing Pattern is used for instruments where the repricing is determined by the 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.

Figure 20-5 Define Relative Repricing Pattern



The steps to create relative repricing patterns are similar to creating absolute repricing patterns.

The only difference is that the fields in the Repricing Events table are different.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Prerequisites

Selecting Relative as the pattern type.

Procedure:

This table describes key terms used for this procedure.

Table 20-2 Key Terms used in Relative Repricing Pattern

Key Terms	Description
Add Row	This allows you to Add one or more Repricing Events.
Add Multiple Rows	Allows you to add more Repricing Events.
Move Up	Allows you to move a particular row up by one position. This action for the first and second rows is not active.
Move Down	Allows you to move a particular row down by one position. This action for the first and last rows is not active.
Delete	This allows you to delete specific rows in the Repricing Events Table. You need to specify the following parameters in the Repricing Events Table for a Relative Repricing Pattern:
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: Days Months Years
Repeat	This allows you to specify the number of times a repricing event should be repeated.
Repricing Method	A drop-down list, it displays the Repricing Type, Flat Rate or Indexed Rate, associated with a particular event.

After defining the above details, click Define. The Event summary Page is displayed. Note the status indicator turns from Red to Green indicating that details for the event have been defined.

The steps to create relative Repricing Patterns are similar to creating Creating Absolute Repricing Patterns.

The only difference is that the fields in the Repricing Events Table are different.

Select Pattern Type as Relative and follow the steps mentioned in Creating Absolute Repricing Patterns section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

21

Transfer Pricing Rules

Transfer Pricing Rules allow you to specify methodologies for Transfer Pricing your product portfolio. A Transfer Pricing rule may contain a Transfer Pricing Methodology defined for a particular product or a set of methodologies defined for all products (Dimension Members) in a particular product hierarchy. In addition, it contains certain parameters used in defining option cost methodologies.

The Transfer Pricing rule is a key component of the Transfer Pricing Process. 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 your Transfer Pricing Rules.

> **Note**
>
> If Transfer Pricing Assumptions are managed at the Parent / Node level, new product dimension members will automatically inherit TP Method assignments.

If new members are added to the applicable product dimension, you need to update your Transfer Pricing Rules by defining appropriate methodologies for the new products.

The procedure for working with and managing the Transfer Pricing Rule is similar to that of other Oracle Funds Transfer Pricing assumption rules. It includes the following steps:

- Searching for Transfer Pricing Rules
- Creating Transfer Pricing Rules
- Viewing and Editing Transfer Pricing Rules
- Copying Transfer Pricing Rules
- Deleting Transfer Pricing Rules

As part of creating and editing Transfer Pricing Rules, you can also define Transfer Pricing Methodologies. See:

- Defining Transfer Pricing Methodologies
- Defining the Redemption Curve Methodology
- Defining the Unpriced Account Methodology

Oracle Funds Transfer Pricing provides you with the option to copy, in total or selectively, the product assumptions contained within the Transfer Pricing, Prepayment, and Adjustments Rules from one currency to another currency or a set of currencies.

- Creating Transfer Pricing Rules
  You create a Transfer Pricing Rule to map Transfer Pricing Methodologies for your products.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Defining Transfer Pricing Methodologies
  The assignment of Transfer Pricing Methodologies is part of the Create or Edit Transfer Pricing Rules process where assumptions about Transfer Pricing Methodologies are made for product-currency combinations.
- Defining Assumptions with the Default Currency
  For cases where you have the same assumption (method and IRC) which is applicable to all currencies or multiple currencies, you can define rules for the combination of Product and Default Currency. To define assumptions for the Default Currency, select a Product from the Hierarchy and Default Currency from the currency list and proceed with the assumption definition. When processing data, the TP engine will first look for an assumption that exactly matches the product or currency of the instrument record. If not found, the engine will then look for the combination of the product and the Default Currency. This is a useful option to utilize during setup when the same product exists across multiple currencies and shares the same TP assumption and Interest Rate Code.
- Defining the Redemption Curve Methodology
  As part of the process for defining the Redemption Curve Methodology, you must select as many Term Points from your selected Transfer Pricing Yield curve as are needed and allocate the percentage weighting for each of those points. The prerequisite for defining the Redemption Curve Methodology is performing basic steps for creating or updating a Transfer Pricing Rule.
- Copying Assumptions across Currencies
  This functionality provides you with the option to copy, in total or selectively, the product assumptions contained within the Transfer Pricing, Prepayment, and Adjustment Rules from one currency to another currency or a set of currencies.

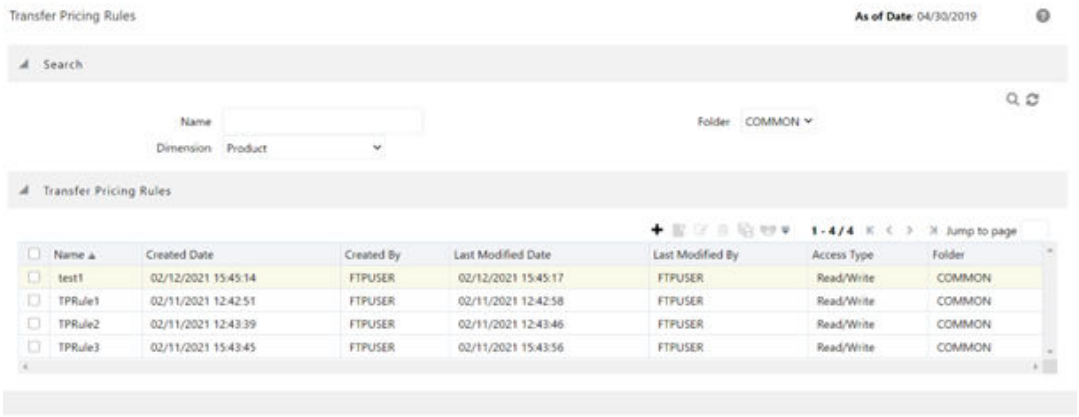
21.1 Creating Transfer Pricing Rules

You create a Transfer Pricing Rule to map Transfer Pricing Methodologies for your products.

To create a Transfer Pricing Rule:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Assumption Specification**, and then select **Transfer Pricing Rules**.

Figure 21-1 Transfer Pricing Rules summary screen



2. Complete standard steps for this procedure.

   In addition to the standard steps for creating rules, the procedure for creating a Transfer Pricing Rule 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The hierarchical relationship between the nodes allows the inheritance of methodologies from Parent nodes to Child nodes.

21.2 Defining Transfer Pricing Methodologies

The assignment of Transfer Pricing Methodologies is part of the Create or Edit Transfer Pricing Rules process where assumptions about Transfer Pricing Methodologies are made for product-currency combinations.

When you click Save in the Create Transfer Pricing Rules process, the rule is saved and the Transfer Pricing Rule summary page is displayed. However, the Transfer Pricing Methodology has not yet been defined for any of your products at this point. You start defining your methodologies for the product-currency combinations before clicking Save. The Transfer Pricing Rule supports the definition of assumptions for combinations of two dimensions: Product and Currency.

You can define Transfer Pricing Methodologies for your entire product portfolio one currency at a time. Suppose your portfolio is comprised of products denominated in two currencies (US Dollar and Japanese Yen) and that you want to specify different Transfer Pricing assumptions and/or different Transfer Pricing yield curves, for each product group. Using the currency selection drop-down list, you can first define assumptions for the products denominated in US Dollars and then proceed with defining assumptions for the Yen-based products.

After you have created a Transfer Pricing rule, you can assign Transfer Pricing Methodologies to product-currency combinations in either of the following two ways:

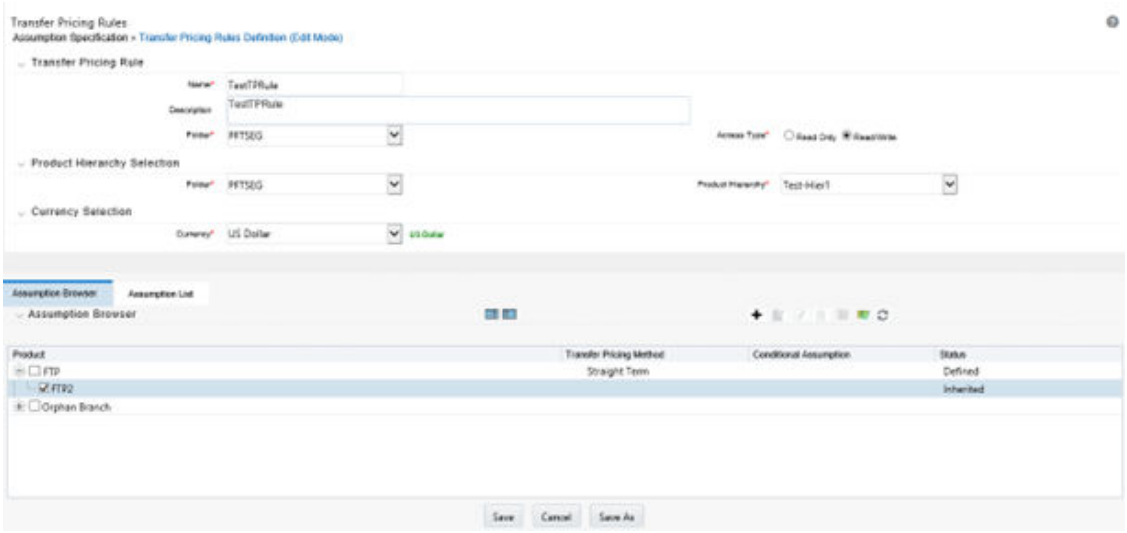
- By creating a conditional assumption using conditional logic. For more information, see Associating Conditional Assumptions with Assumption Rules.
- Directly on the Transfer Pricing Methodology page, as described here.

21.3 Defining Assumptions with the Default Currency

For cases where you have the same assumption (method and IRC) which is applicable to all currencies or multiple currencies, you can define rules for the combination of Product and Default Currency. To define assumptions for the Default Currency, select a Product from the Hierarchy and Default Currency from the currency list and proceed with the assumption definition. When processing data, the TP engine will first look for an assumption that exactly matches the product or currency of the instrument record. If not found, the engine will then look for the combination of the product and the Default Currency. This is a useful option to utilize during setup when the same product exists across multiple currencies and shares the same TP assumption and Interest Rate Code.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 21-2 Transfer Pricing Rule Definition (Edit Mode)



Default Currency setup example: If you have two instrument records of the same product, each with a different currency, for example, 1 is 'USD' and the other is 'AUD', you have two configuration choices. You can either:

- Define the assumptions individually for each product-currency combination using direct input or copy across.
- You can create 1 assumption for the combination of Product and *Default Currency*. When you use *Default Currency*, the TP Engine will apply this assumption to ALL currencies (unless a direct assumption is available for the product + currency processed). In the case where users have many individual currencies that utilize the same TP Method and reference IRC rates, this is a useful option because you only have to define the assumption 1 time and it applies to many different Product + Currency combinations.

For defining assumptions with Default Currency, you must perform the basic steps for creating or updating Transfer Pricing Rules.

Procedure:

Table 21-1 Fields in the Transfer Pricing Rule Definition screen

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.
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 is used to generate transfer rates in the Spread from Interest Rate and Spread from Note Rate methods.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 21-1 (Cont.) Fields in the Transfer Pricing Rule Definition screen

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 selected while modeling the effect of serviced portfolios where the underlying assets are 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.
Audit Trail	Select to generate Audit Trail Output for specific product/ currency combination.
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 use as source values when you transfer price using the Unpriced Account Method.
Across All Organization Units	When this option is enabled, the Transfer Price is calculated as a weighted average across all organization units for the matching product value and currency, and any optional migration dimensions selected in the Transfer Pricing Process Rule. Otherwise, the Transfer Price is calculated from accounts only within a particular Organizational Unit.
Holiday Calendar	Holiday Calendars are defined in the Holiday Calendars UI. In the Holiday Calendar, you can specify weekend days and Holiday Dates as applicable.
Rolling Convention	Rolling Conventions allow you to specify how dates falling on specified weekends or holidays should be handled.
Interest Calculation Logic	The Interest Calculation Logic Assumption allows you to specify whether to simply the date of the computed cash flow or to shift the date and recalculate the interest payment amount.

To define the assumptions with the default currency:

1. Navigate to the Assumption Browser page.
2. Select a **Product Hierarchy**.
3. Select a **Currency**.
   The list of currencies available for selection is managed within Rate Management and reflects the list of Active currencies.

Copyright © 2011, 2026, Oracle and/or its affiliates.

4. Expand the hierarchy and select one or more members (leaf values and/or node values) from the product hierarchy.
5. Click the **Add** icon to begin mapping Transfer Pricing Methods to the list of selected product dimension members. The system displays a list of all the products (for which you can define assumptions) or currencies (that are active in the system).
6. From the Transfer Pricing Method Selector page, select the appropriate data source: **Account Tables** or **Ledger Table**.
7. Select the Transfer Pricing Method for the selected product member.

> **Note**
>
> The Transfer Pricing Methodologies available depend on the selected data source. See: Transfer Pricing Combinations.

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. See also:

- Defining the Redemption Curve Methodology
- Defining the Unpriced Account Methodology

8. Select **Output Audit Trail** to output the audit data at the time of processing.
9. Specify the desired **Option Cost** Methodology. This option is available only when the data source is Account Tables. You can specify an Option Cost Methodology as follows:
   a. Select Run using Monte Carlo Option Cost Method. The Target Balance drop-down list is displayed.
   b. 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

> **Note**
>
> The Market_Value_c column should be populated if you make this selection, either by an Oracle ALM process or via direct load from an external source.

For more information on Transfer Pricing Option Cost, See OFS Cash Flow Engine Reference Guide and Stochastic Transfer Pricing Process and Option Cost Parameters.

10. Select the **Holiday Calendar**. The screen displays the Holiday Calendar inputs only for Cash Flow TP methods – Duration, Average Life, Weighted Term, Zero Discount Factors. The default assumption is None, meaning Holiday Calendar adjustments are turned off. If a Holiday Calendar is selected, Holiday Calendar adjustments will be enabled and the following two additional inputs will be required:
    - **Rolling Convention**
      – **Following Business Day**: The payment date is rolled to the next business day.

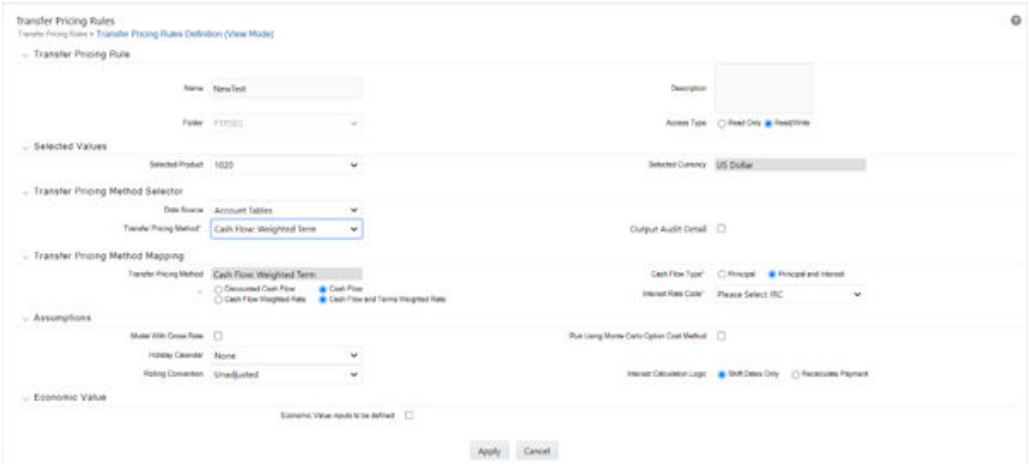
Copyright © 2011, 2026, Oracle and/or its affiliates.

  - **Modified following Business Day**: The payment date is rolled to the next business day unless doing so would cause the payment to be in the next calendar month, in which case the payment date is rolled to the previous business day.
  - **Previous Business Day**: The payment date is rolled to the previous business day.
  - **Modified previous Business Day**: The payment date is rolled to the previous business day unless doing so would cause the payment to be in the previous calendar month, in which case the payment date is rolled to the next business day

- **Interest Calculation Logic**
  - **Shift Dates Only**: If a future payment date (as computed by the Cash Flow Engine (CFE)) falls on a designated holiday (including weekends), the CFE will shift the payment date from the holiday as per the rolling convention. No changes will be made to the payment amount or accrual amount; this is simply shifting the date on which the cash flow will post. The subsequent payment dates resume according to the original schedule.
  - **Recalculate Payment**: This option includes the same Holiday Calendar definition as in the Shift Dates Only option, but it also takes one additional step to recalculate the interest payment amount (and interest accruals) based on the actual number of days in the (adjusted) payment period. The instrument records use the payment frequency (term and multiplier) and the Re-Price frequency (term and multiplier) in association with the Next/Last Payment Date and Next/Last Re-Pricing Date to determine when the cash flow will post. The CFE logic is enhanced to acknowledge holiday dates and re-compute the payment/interest amount given the change in days. In addition, the engine gets back on the scheduled track of payment events after a holiday event occurs in one (or many sequential) events.

> **Note**
>
> Holiday Calendar adjustments can also be applied to the Tractor TP Method. For this method, the Holiday Calendar assumptions are defined within the Behavior Pattern > Replicating Portfolio UI.

Figure 21-3 Transfer Pricing Rule Definition (Edit Mode)



11. Select **Economic Value**. This section displays the inputs required for calculating Economic Value. These assumptions are optional and the section appears only when the “Economic Value inputs to be defined” check box is selected.
    The following inputs are required for calculating Economic Value:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Cash Flow Interest Type**: Select the interest rate to use for calculating the Interest Cash Flow. This interest amount, together with the principal amount will be discounted and used to arrive at the Economic Value of the instrument Record.
- **Interest Only**: Select this option if you want to exclude the Principal Cash Flow from the Economic Value Calculation.
- **Exclude Accrued Interest**: Select this option if you want to exclude accrued interest, (interest computed from last payment date to As-of-Date) from the Economic Value calculation. This will provide you with a clean price.
- **Interest Rate Code**: Select the Interest Rate code to be used for discounting the cash flows.
- **Interest Rate Spread**: Input any applicable spread to be added on top of the IRC rate.

> **Note**
>
> Economic value calculations are now supported for account rate tiers as well.

12. Click **Apply**.
    At this point, you can:
    - Continue defining additional methodologies for other product-currency combinations contained in your selection set, by repeating the above procedure.
    - Complete the process by clicking Cancel or by answering NO to the confirmation alert after applying the assumptions for each Product or Currency combination in your selected set.
13. From the Assumption Browser page, click **Save**.
    The new assumptions are saved and the Transfer Pricing Rule Selector page is displayed.

> **Note**
>
> Oracle Funds Transfer Pricing provides you with the option to copy, in total or selectively, the product assumptions contained within the Transfer Pricing, Prepayment, and Adjustment rules from one currency to another currency or a set of currencies. For more information, see Copying Assumptions Across Currencies.

- Availability of Transfer Pricing Methodologies
  The availability of Transfer Pricing Methodologies depends on the data source that you select: Account Table or 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.
- Required Parameters
  You cannot define a Transfer Pricing Methodology successfully unless you specify the required parameters.

21.3.1 Availability of Transfer Pricing Methodologies

The availability of Transfer Pricing Methodologies depends on the data source that you select: Account Table or 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Interest Rate Code LOV is filtered by the selected Currency.

Table 21-2 Transfer Pricing Combinations

Transfer Pricing Methodology	Data Source: Account Table	Data Source: Ledger Table	Interest Rate Code	Behavior Pattern	Holiday Calendar
Do Not Calculate	Yes	Yes			
Cash Flow: Average Life	Yes		Yes		Yes
Cash Flow: Duration	Yes		Yes		Yes
Cash Flow: Weighted Term	Yes		Yes		Yes
Cash Flow: Zero Discount Factors	Yes		Yes		Yes
Moving Averages	Yes	Yes	Yes		
Straight Term	Yes		Yes		
Spread from Interest Rate Code	Yes	Yes	Yes		
Spread from Note Rate	Yes				
Redemption Curve	Yes	Yes	Yes		
Tractor Method	Yes	Yes	Yes	Yes	Yes
Caterpillar	Yes	Yes	Yes	Yes	
Weighted Average Perpetual	Yes	Yes	Yes	Yes	
Unpriced Account		Yes			

> **Note**
>
> Not specifying assumptions for a node is not the same as selecting the Do Not Calculate methodology. Child nodes for which no assumptions are specified automatically inherit the methodology of their closest parent node. Therefore, if neither a Child node nor its immediate Parent has a method assigned, the application searches up the nodes in the hierarchy until it finds a Parent node with a method assigned, and uses that method for the Child node. However, if no Parent node has a method assigned then the application triggers a processing error stating that no assumptions are assigned for the particular product/currency combination. However, if the Parent node has the method *Do Not Calculate* assigned to it then the Child node inherits *Do Not Calculate,* obviating the need for calculation and for a processing error.

Copyright © 2011, 2026, Oracle and/or its affiliates.

21.3.2 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.

Table 21-3 Required Parameters for a Transfer Pricing Methodology

Transfer Price Method	Yield Curve Term	Historical Range	Lag Term	Rate Spread	Assignment Date	Mid Period	Term Points	Dimension Values
Cash Flow: Average Life								
Cash Flow: Weighted Term								
Cash Flow: Duration								
Cash Flow: Zero Discount Factors								
Moving Averages	Required	Required						
Straight Term						Optional		
Tractor Method								
Spread from IRC	Required		Required	Required	Required	Optional		
Spread from Note Rate				Required		Optional		
Redemption Curve					Required	Optional	Required	
Caterpillar								
Do not Calculate								
Weighted Average Perpetual								
Unpriced Account								Required
Tractor Method								

Copyright © 2011, 2026, Oracle and/or its affiliates.

21.4 Defining the Redemption Curve Methodology

As part of the process for defining the Redemption Curve Methodology, you must select as many Term Points from your selected Transfer Pricing Yield curve as are needed and allocate the percentage weighting for each of those points. The prerequisite for defining the Redemption Curve Methodology is performing basic steps for creating or updating a Transfer Pricing Rule.

To add the term steps:

1. Click **Add New Term Points** to display the Add New Term Points page.
2. Select the **Transfer Pricing Yield Curve Points** as required.
3. Click **OK**.

   The Term Point Selection section 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.

5. To remove a Yield Curve Point from the Percentages/Term Points table, select the term point(s) and click the **Delete** icon.

- Defining Tractor Methodology
  The prerequisites for defining a tractor methodology are:
- Defining the Unpriced Account Methodology
  When defining an Unpriced Account Methodology, you need to select the Product dimension members (products) whose weighted average transfer rate will be assigned to the product or currency combination being defined. The prerequisite for defining the Unpriced Account Methodology is performing basic steps for creating or upgrading a Transfer Pricing Rule.

21.4.1 Defining Tractor Methodology

The prerequisites for defining a tractor methodology are:

- Creating a Non-Maturity Behavior Pattern with the Replicating Portfolio option enabled.
- Generating (and maintaining) the Portfolio. Volatile and Core Instrument strips will be created in the FSI_M_REP_PORTFOLIO_STRIPS table.

To define and use a tractor TP method:

1. Define the Transfer Pricing Rule and select the Tractor Method from the list of available TP Methods for relevant Product dimension members.

   (Optional) <Enter a step example.>
2. Select the appropriate Behavior Pattern.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The list will be limited to only those Behavior Patterns that are defined as Replicating Portfolios.

3. Select the Transfer Pricing Interest Rate Code.
4. Define a Transfer Pricing Process and Run using the TP Rule.
   - TP Process transfers price the non-zero portfolio strips using a straight term method.
   - TP Process computes a weighted average TP rate for the portfolio and will update all instruments mapped through the TP rule to this method.

21.4.2 Defining the Unpriced Account Methodology

When defining an Unpriced Account Methodology, you need to select the Product dimension members (products) whose weighted average transfer rate will be assigned to the product or currency combination being defined. The prerequisite for defining the Unpriced Account Methodology is performing basic steps for creating or upgrading a Transfer Pricing Rule.

To add the dimension values:

1. Click the **Dimensional Values** icon to display the Hierarchical Add Members page.
2. Search and select the required dimension members. Specify whether the weighted average of Transfer Rates has to be taken across all organizational units or for accounts only within that organizational unit.

> **Note**
>
> You must also select the Organization Unit dimension along with any other applicable dimensions under Migration Dimensions on the Migration tab of the TP Process when using this method.

3. Click **Apply**.

   The Transfer Pricing Assumption Browser page is displayed.

21.5 Copying Assumptions across Currencies

This functionality provides you with the option to copy, in total or selectively, the product assumptions contained within the Transfer Pricing, Prepayment, and Adjustment Rules from one currency to another currency or a set of currencies.

Copy of assumptions across currencies enhances the usability of Oracle Funds Transfer Pricing in a multi-currency environment. For example, if you have 10 currencies enabled in the application, you need to input only one set of assumptions and then copy those assumptions across all enabled currencies, instead of having to input 10 full sets, thereby saving a significant amount of input time.

This functionality also reduces the risk associated with data input errors, as you need to audit inputs for a single set of assumptions before executing the copy procedure. The copy across the currencies process requires users to select a replacement Transfer Pricing yield curve for each target currency. These currency-specific IRC's replace the IRC selection made for each product in the source currency selection set.

Copyright © 2011, 2026, Oracle and/or its affiliates.

You must define Transfer Pricing, Prepayment, and (or) Adjustment Rules related to product assumptions.

To copy the assumptions across currencies:

1. Navigate to the appropriate (Transfer Pricing, Prepayment, or Adjustment rule) Assumption Browser.
2. Select **Source currency**.
3. Select defined product assumptions individually using the checkboxes corresponding to each product (or Node on the hierarchy).
4. Click the **Copy Across Currencies** icon.
5. On the Copy Across Currencies page, select the listed currencies either individually using the corresponding checkboxes or in total using **Select All**.
6. Specify an interest rate code for each selected currency. This is necessary because each interest rate code is specific to a single currency. When copying product assumptions across currencies, you must define the interest rate code for each target currency to replace the interest rate code used for the source currency assumptions. For Transfer Pricing Rules that use the Redemption Curve Method, users should pay careful attention to the structure of the Interest Rate Codes selected for the Target currencies to ensure they contain all of the Term Points used in the definition of the source assumptions. If the selected target Interest Rate Code structures are missing required Term Points, the UI will display a notification regarding the missing Term Points, and assumptions cannot be copied until the user takes corrective action.
7. Click **Apply** to initiate the copy process and to return to the Assumption Browser page.

> **Note**
>
> You can review the results of the copy process from the Assumption Browser by selecting a different currency and following the usual navigation to view or edit assumptions. The application displays new assumptions for each product that was included in the original source selection. The copy process replaces pre-existing assumptions for any product-currency combination that is included in the target selection.

8. Click **Save** on the Assumption Browser page to save the assumptions to the database.

> **Note**
>
> Save or Cancel operation on assumption browser page is only applicable to top half of the page. Changes which are already committed on the assumption browser, like method addition/deletion or conditional assumptions are already committed actions which will remain intact regardless of user clicking save/cancel on assumption browser page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

22

Prepayment Rules

This module describes the procedure for working with and managing Prepayment Rules. Prepayment rules allow you to specify methodologies to model the loan prepayment and deposit early redemption behavior of products in your portfolio and quantify the associated prepayment risk in monetary terms.

Topics:

- Search Prepayment Rule. For more information, see the Searching for Rules section.
- Create Prepayment Rules
- View and Edit Prepayment Rule. For more information, see the View and Edit the Rule section.
- Copy Prepayment Rule. For more information, see the Copy the Rule section
- Delete Prepayment Rule. For more information, see the Delete the Rule section

The methodologies contained in the Prepayment rule are referenced by Transfer Pricing and ALM Processes. The procedure for working with and managing the Prepayment rule is similar to that of other Oracle Asset Liability Management business rules.

Oracle Asset Liability Management provides you with the option to copy, in total or selectively, the product assumptions contained within the Prepayment rules from one currency to another currency or a set of currencies or from one product to another product or a set of products.

- Create Prepayment Rules
- Defining Prepayment Methodologies

22.1 Create Prepayment Rules

You create a Prepayment Rule to define prepayment assumptions for new products.

To create a new Prepayment Rule, follow these steps:

- Navigate to the Prepayment rule summary page.
- Complete standard steps for this procedure.

> **Note**
>
> In addition to the standard steps for creating rules, the procedure for creating a Prepayment rule involves one extra step. After Standard Step 6, 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 child nodes.

Copyright © 2011, 2026, Oracle and/or its affiliates.

22.2 Defining Prepayment Methodologies

The assignment of prepayment assumptions is part of the Create or Edit Prepayment Rule Process where assumptions about loan prepayments or deposit early redemptions are made for product-currency combinations. When you click Save in the Create Prepayment Rules Process, the Rule is saved and the Prepayment Rule Summary Page is displayed. However, prepayment assumptions have not yet been defined for any of your products at this point. Typically, you would start defining your prepayment assumptions for product-currency combinations before clicking Save.

The Prepayment Rule supports the definition of prepayment assumptions for combinations of two dimensions: Product and Currency.

Once you have created a Prepayment Rule, you can assign prepayment methodologies to product-currency combinations using Node Level Assumption.

- By creating a conditional assumption using conditional logic. For more information, see Associating Conditional Assumptions with Prepayment Rules section.
- Directly on the Prepayment methodology page, as described here.
- Defining Assumptions with the Default Currency
  For cases where you have the same assumption (method and IRC) which is applicable to all currencies or multiple currencies, you can define rules for the combination of Product and Default Currency.
- Defining Constant Prepayment Method
- Defining Prepayment Model Method
- Defining PSA Prepayment Method
- Defining the Arctangent Calculation Method
- Defining Early Redemption Assumptions

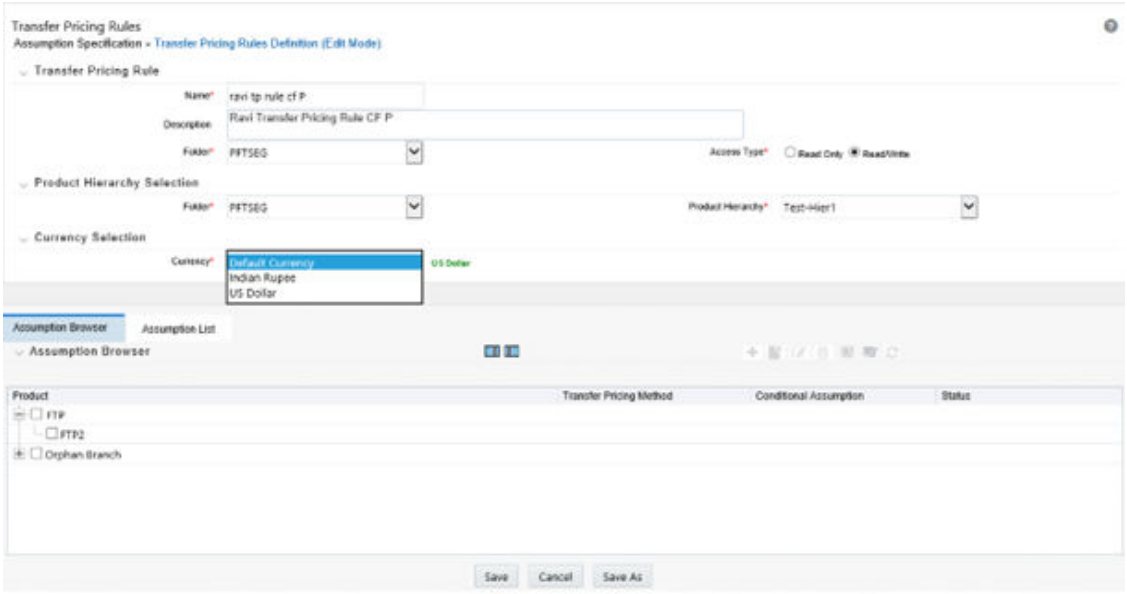
22.2.1 Defining Assumptions with the Default Currency

For cases where you have the same assumption (method and IRC) which is applicable to all currencies or multiple currencies, you can define rules for the combination of Product and Default Currency.

To define assumptions for the Default Currency, select a Product from the Hierarchy and Default Currency from the currency list and proceed with the assumption definition as described above. When processing data, the TP engine will first look for an assumption that exactly matches the product or currency of the instrument record. If not found, the engine will then look for the combination of the product and the Default Currency. This is a useful option to utilize during setup when the same product exists across multiple currencies and shares the same TP assumption and Interest Rate Code.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 22-1 Transfer Pricing Rules - Definition Mode



Default Currency setup example: If you have two Instrument Records of the same Product, each with a different currency, for example, 1 is 'USD' and the other is 'AUD', you have two configuration choices. You can either:

- Define the assumptions individually for each product-currency combination using direct input or copy across.
- You can create one assumption for the combination of Product and “Default Currency”. When you use “Default Currency”, the TP Engine will apply this assumption to ALL currencies (unless a direct assumption is available for the product + currency processed). In the case where users have many individual currencies that utilize the same TP Method and reference IRC rates, this is a useful option because you only have to define the assumption 1 time and it applies to many different Product + Currency combinations.

22.2.2 Defining Constant Prepayment Method

Use this procedure to define prepayment assumptions using the 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.

Prerequisites

Performing basic steps for creating or updating a Prepayment Rule.

Procedure

Users also have two options for determining the timing of the Constant Prepayment assumption. The options include:

- **Use Payment Dates:** This is the default option. If this option is selected, then Constant Prepayment Runoff will occur on scheduled payment dates only.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **User Defined Prepayment Tenors:** If this option is selected, users can specify any runoff timing. For example, users might choose to define the prepayment to the Runoff on the first day of the forecast.

The above options will be available only for Asset Instrument types.

To define constant prepayment within the Prepayment Rule, follow the steps given in below sections:

- Use Payment Dates
- User Defined Prepayment Tenors

Use Payment Dates

1. Select the **Use Payment Dates** Option.
2. Select the **Payment Event Type** Option.
3. Select the **Start Origination Date** using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.
   The first cell in the Start Origination Date Column and all the cells in the End Origination Date Column are read-only. This ensures that all possible origination dates must support reference values when Prepayment assumption lookups occur.

   Each row in the End Origination Date Column is filled in by the system when you click **Add Row** or save the Rule.

   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 before the previous End Origination Date Field.
4. 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.
5. The **Percent** column represents the actual annualized prepayment percentage that the system uses to generate the principal runoff during the Cash Flow calculations.
6. Click **Add Row** to add additional rows and click the corresponding **Delete** button to delete a row.
7. You can add as many rows as possible in this table using **Add Multiple Row** Option. However, you need to enter relevant parameters for each new row.
8. You can also use the **Download Excel** feature to export the Prepayment rate information that is displayed on screen, modify, and copy-paste it back in the grid.
9. Define Seasonality assumptions if required to model date-specific adjustments to the annual prepayment rate. Inputs act as a multiplier, For Example, an input of 2 will double the prepayment rate in the indicated month.

User Defined Prepayment Tenors

1. Select the **User Defined Prepayment Tenors** Option. This option allows you to specify the term and multiplier to the Prepayment Date for the row. The term is used to derive Prepayment Dates with reference to As of Date.
2. You can calculate the prepayment rate based on Current/Reducing Balance and Annual/De-annual Prepayment Rate.
3. Select the **Balance Type** as **Current Balance** or **Reducing Balance**.
   - If the Balance Type is selected as Current Balance, then the prepayment amount will be calculated using Principal Balance on As of Date. That is, without reducing the balance by any payment/prepayment that may have occurred between as of the date and prepayment date.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- If the Balance Type is selected as Reducing Balance, then the prepayment amount will be calculated using balance as on Prepayment Date. That is, after reducing the Principal Balance by any payment/prepayment that may have occurred between as of date and prepayment date.

4. Select the Prepayment Rate Type as Annual Prepayment Rate or De-annual Prepayment Rate.
   When the Annual Prepayment Rate is selected then the prepayment rate entered in the screen is directly used.

   In the other case, the rate entered in the screen is de-annualized before calculating the prepayment amount.
5. Enter the **Start Origination Date** and **End Origination Date** ranges, add additional ranges as required using the Add Row button.
6. Enter the term to Runoff Tenor and Multiplier for each of the date ranges.
7. Enter the **Annual Prepayment Rate Percent** for each of the date ranges.
8. Enter 'Repeat' if you want the same prepayment to occur multiple times. By default, it is set to 1.
9. Click **Add Row** to add additional runoff % rows and click the corresponding **Delete** button to delete a row.
   You can add as many rows as possible in this table using **Add Multiple Row** Option. However, you need to enter relevant parameters for each new row.
10. Define Seasonality assumptions as required to model date specific adjustments to the annual Prepayment Rate. Inputs act as a multiplier, for example, an input of 2 will double the Prepayment Rate in the indicated month.

The manner in which prepayment / redemption tenor is interpreted has changed from ALM 8.0.7 onwards in case of user defined prepayment and user defined early redemption. Earlier runoff occurred on “As of date + Prepayment / early redemption tenor”. From ALM 8.0.7 onwards only first runoff occurs on “As of date + Prepayment / early redemption tenor”. Subsequent runoff occur on “Previous prepayment / early redemption date + Prepayment / early redemption tenor”. Thus customers who are upgrading or migrating to ALM 807 from a previous release must appropriately modify tenors in prepayment rules where user defined prepayment tenor and user defined early redemption tenor have been used.

22.2.3 Defining Prepayment Model Method

Use this procedure to define prepayment assumptions using the Prepayment Model Calculation method. The Prepayment Model Method allows you to define more complex prepayment assumptions compared to the other Prepayment Methods. Under this method, prepayment assumptions are assigned using a custom Prepayment Model. You can build a Prepayment model 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 rate.

Prerequisites

- Prepayment Model must be created.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Performing basic steps for creating or updating a Prepayment Rule.

Procedure

Users also have two options for determining the timing of the Prepayment Model assumption. The options include:

1. Define the source for the Market Rate by Selecting an Index (Interest Rate Code) from the list of values.
2. Enter the Spread. The spread is added to the rate from the underlying interest rate curve to determine the market rate.
3. Select an **Associated Term**: Remaining Term, Reprice Frequency, or Original Term.
4. Specify the Prepayment Model parameters.
    - Enter the Coefficient (if needed) by which the Prepayment Rate should be multiplied. This multiple is applied to the instruments for which the origination date lies in the range defined in the Start Origination Date-End Origination Date fields.
    - Select the Start Origination Date using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.
    - Select a predefined prepayment model from the Prepayment model Rule list of values. Click the **View Details** icon to preview the selected Prepayment Model.
    - The system uses the prepayment model assumptions to calculate the prepayment amounts for each period. You need to associate a prepayment model for every Start Origination-End Origination Date range.
    - 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 model as you require. However, you need to enter relevant parameters for each new row. You can also use the Excel import/export feature to add the Prepayment rate information.
5. Define Seasonality assumptions as required to model date specific adjustments to the annual prepayment rate. Inputs act as multiplier, for example, an input of 2 will double the prepayment rate in the indicated month

22.2.4 Defining PSA Prepayment Method

Use this procedure to define Prepayment Assumptions using the PSA Prepayment Method. The PSA Prepayment method (Public Securities Association Standard Prepayment Model) is a Standardized Prepayment Model that is built on a single dimension, expired term. The PSA Curve is a schedule of prepayments which assumes that prepayments will occur at a rate of 0.2 percent CPR in the first month and will increase an additional 0.2 percent CPR each month until the 30th month and will prepay at a rate of 6 percent CPR thereafter ("100 percent PSA"). PSA Prepayment Speeds are expressed as a multiple of this base scenario. For example, 200 percent PSA assumes Annual Prepayment Rates will be twice as fast in each of these periods - 0.4 percent in the first month, 0.8 percent in the second month, reaching 12 percent in month 30 and remaining at 12 percent after that. A zero percent PSA assumes no prepayments. You can create your own Origination Date ranges and assign a particular PSA Speed to all the instruments with origination dates within a particular Origination Date range. PSA Speed inputs can be between 0 and 1667.

Prerequisites

Performing basic steps for creating or updating a Prepayment Rule.

Procedure

Copyright © 2011, 2026, Oracle and/or its affiliates.

Prepayment under this method occurs on Payment Dates only.

1. Select the **Payment Event Type** option.
2. Select the **Start Origination Date** using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.
   The first cell in the **Start Origination Date** Column and all the cells in the **End Origination Date** Column are Read-Only. This ensures that all possible Origination Dates have supporting reference values when Prepayment Assumption Lookups occur. Each row in the End Origination Date Column is filled in by the system when you click Add Row or save the Rule.

   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 before the previous End Origination Date Field.
3. Enter the PSA Speed that you want to apply to the instruments having Origination Dates in a particular Start Origination-End Origination Date range. The PSA Method is based on a standard PSA curve. You can view the seeded model by selecting the View Prepayment Model icon.
   The default value is 100 PSA and inputs can range from 0 to 1667.
4. Click **Add Row** to add additional rows and click the corresponding **Delete** Option to delete a row.
   You can add as many rows as possible in this table using Add Multiple Row Option. However, you need to enter relevant parameters for each new row.
5. You can also use the **Download Excel** Feature to export the Prepayment Rate Information that is displayed on screen, modify, and copy-paste it back in the grid.
6. Define Seasonality Assumptions as required to Model Date specific adjustments to the Annual Prepayment Rate. Inputs act as a multiplier, For example, an input of 2 will double the Prepayment Rate in the indicated month.

22.2.5 Defining the 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). Use this procedure to define Prepayment Assumptions using the Arctangent Calculation Method.

Prerequisites

Performing basic steps for creating or updating a Prepayment Rule.

Procedure

Prepayment under this method occurs on Payment Dates only.

1. Define the source for the Market Rate by Selecting an Index (Interest Rate Code) from the list of values.
2. Enter the Spread.
3. The spread is added to the rate from the underlying interest rate curve to determine the market rate.
4. Select an Associated Term: Original Term, Reprice Frequency, or Remaining Term.
5. Specify the Arctangent Argument table parameters.
6. Select the Start Origination Date using the date picker. Alternatively, you can enter the Start Origination Date in the space provided.

Copyright © 2011, 2026, Oracle and/or its affiliates.

7. 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.
8. Click Add Another Row.
9. You can add as many rows in this table as you require. However you need to enter relevant parameters for each new row.
10. You can also use the Excel import/export feature to add the Prepayment rate information.
11. Define the Seasonality assumptions as required to model date specific adjustments to the annual prepayment rate. Inputs act as multiplier, For example, an input of 2 will double the prepayment rate in the indicated month.

22.2.6 Defining Early Redemption Assumptions

If you are working with deposit products, it is possible to define Early Redemption assumptions within the Prepayment Rule. While defining assumptions, the Prepayment rule will consider whether or not the product is an asset or liability (based on the account type attribute defined in dimension member management). If the product is an asset, then the Prepayments tab will be active in the prepayment assumption detail page. If the product is a liability, then the Early Redemption tab will be active.

Prerequisites

- Performing basic steps for creating or updating a Prepayment rule.
- To define Early Redemption assumptions, the account type for the selected product must be a Liability.

Procedure

The procedure for defining Early Redemptions is the same as noted above for prepayments, with two exceptions:

- The list of Calculation Methods is limited to Constant and Prepayment Models
- The range definitions are based on Maturity Date ranges of the instruments rather than Origination Date ranges

Users also have two options for determining the timing of the early redemption assumption. Options include:

- Use Payment Dates: This is the default option. If selected early redemption runoff will occur on scheduled payment dates only.
- User Defined Redemption Tenors: If selected, users can specify any runoff timing. For example, users might choose to define the early redemption to runoff on the first day of the forecast.

To define Early Redemptions within the Prepayment Rule, follow the steps given below:

Use Payment Dates

1. Select the Use Payment Dates option.
2. Enter the Start Maturity and End Maturity Dates.
3. The first cell in the Start Maturity Date column and all of the cells in the End Maturity Date column are read only. This ensures that all possible Maturity dates have supporting reference values when Prepayment assumption lookups occur. Each row in the End Maturity Date column is filled in by the system when you click Add Row or save the rule. The first Start Maturity Date (in row 1) has a default value of January 1, 1900. When you

Copyright © 2011, 2026, Oracle and/or its affiliates.

enter a Start Maturity Date in the next row, the system inserts a date that is a day prior to the previous End Maturity Date field.

4. Enter the annual prepayment rate percent that you want to apply to the instruments having origination dates in a particular Start Maturity-End Maturity Date range.
5. The Percent column represents the actual annualized prepayment percentage that the system uses to generate the principal runoff during the cash flow calculations.
6. Click Add Row to add additional rows and click the corresponding Delete icon to delete a row.
7. You can add as many rows in this table as you require. However you need to enter relevant parameters for each new row.
8. You can use Data Input Helper feature. For more information, see Data Loader section.
9. You can also use the Excel import/export feature to add the Prepayment rate information. For more details, see Excel Import/Export section.

User Defined Redemption Tenors

1. Select the User Defined Redemption Tenors option. This option allows you to specify the term to runoff for the particular row.
2. Enter the Start Maturity and End Maturity date ranges, add additional ranges as required using the Add Row button.
3. You can calculate the prepayment rate based on Current/Reducing Balance and Annual/De-annual Prepayment Rate. Select the Balance Type as Current Balance or Reducing Balance.
4. If the Balance Type is selected as Current Balance, then the prepayment amount will be calculated using CUR_PAR_BAL on As of Date. That is, without reducing the balance by any payment/prepayment that may have occurred between as of date and prepayment date.
5. If the Balance Type is selected as Reducing Balance, then the prepayment amount will be calculated using balance as on Prepayment Date. That is, after reducing the CUR_PAR_BAL by any payment/prepayment that may have occurred between as of date and prepayment date.
6. Select the Prepayment Rate Type as Annual Prepayment Rate Or De-annual Prepayment Rate.
7. When Annual Prepayment Rate is selected then prepayment rate entered in screen is directly used. In the other case, rate entered in screen in de-annualised before calculating prepayment amount.
8. Enter the Start Maturity Date and End Maturity Date ranges, add additional ranges as required using the Add Row button.
9. Enter the term to runoff tenor and multiplier for each of the date ranges.
10. Enter the early redemption runoff percentage for each of the date ranges.
11. Enter 'Repeat' if you want same prepayment to occurs multiple times. By default it is set to 1.
12. Click Add Row Delete.
13. Define Seasonality assumptions as required to model date specific adjustments to the annual prepayment rate or early redemption rate. Inputs act as multiplier, for example an input of 2 will double the runoff rate in the indicated month.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The manner in which prepayment / redemption tenor is interpreted has changed from ALM 8.0.7.0.0 onwards in case of user defined prepayment and user defined early redemption. Earlier runoff occurred on “As of date + Prepayment / early redemption tenor”. From ALM 8.0.7.0.0 onwards only first runoff occurs on “As of date + Prepayment / early redemption tenor”. Subsequent runoff occur on “Previous prepayment / early redemption date + Prepayment / early redemption tenor”. Thus customers who are upgrading or migrating to ALM 8.0.7.0.0 from a previous release must appropriately modify tenors in prepayment rules where user defined prepayment tenor and user defined early redemption tenor have been used.

Copyright © 2011, 2026, Oracle and/or its affiliates.

23
Prepayment Models

This module describes the procedure to build Prepayment Models. These Prepayment Models can be referenced by a Prepayment Rule to Model Prepayment Behavior of instruments based on a range of instrument level attributes.

The Prepayment Model consists of the Prepayment Dimensions and the Bucket Values for these Dimensions. To define the Prepayment Model Structure, you can select a maximum of three prepayment dimensions. After the dimensions and the number of buckets (tiers) are defined, you need to assign values to the buckets.

The procedure for working with and managing Prepayment Models is similar to that of other assumption rules.

Topics:

- Search Prepayment Model Rule. For more information, see the Searching for Rules section.
- Create Prepayment Models
- View and Edit Prepayment Model Rule. For more information, see the View and Edit the Rule section.
- Copy Prepayment Model Rule. For more information, see the Copy the Rule section
- Delete Prepayment Model Rule. For more information, see the Delete the Rule section
- Create Prepayment Models

23.1 Create Prepayment Models

Creating a Prepayment Model comprises the following sub procedures:

1. Creating Prepayment Models
2. Defining the structure of the Prepayment Model
3. Assigning Node Values

You can create Prepayment Models with following Rate Calculation options:

- Manual
- External Model
- Integrated Model

This table describes key terms used for this procedure.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 23-1 Fields to create Prepayment Model

Term	Description
Dimension	Influences the prepayment behavior of an instrument. You can build a prepayment model using up to three prepayment dimensions. Each dimension maps to an attribute of the underlying transaction (For example, age/term or rate and so on) so the cash flow engine can apply a different prepayment rate based on the specific characteristics of the record.
Lookup method	Used to calculate prepayment rates for the prepayment dimension values that do not fall exactly on the defined prepayment dimension nodes. Oracle Asset Liability Management offers the following lookup methods: • **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 • **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. The following example explains the differences between these two lookup methods. The following lists show the age and corresponding prepayment rates of instruments. **Age** 12 24 36 60 **Prepayment Rates** 5 10 15 20 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, 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 24 months to 35.9999 months.
Nodes	Exact points for each dimension where attribute information has been defined.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 23-1 (Cont.) Fields to create Prepayment Model

Term	Description
Prepayment Rate Calculation	Select Manual, Integrated Model, or External models. For more information, see the Prepayment Rate Calculation section. Note: Only 'Manual' option is available to in CFE.

1. Navigate to the Prepayment Model summary page.
2. Complete standard steps for this procedure. For more information, see Creating Rules section.

Figure 23-1 Prepayment Model



3. In case of ALM/FTP/BSP, select prepayment rate calculation which could either be manual, integrated or external model.

- Defining the Structure of the Prepayment Model
- Prepayment Rate Calculation
- Editing Prepayment Models
- Editing Prepayment Rates in a Prepayment Model
- Editing Prepayment Rate Calculation
  As part of editing Prepayment Rate Calculation, you can change the Prepayment Rate Calculation and the structure of the Prepayment Model.
- Prepayment Rate Data Loader
- Prepayment Model Analysis
- Prepayment Rate Calculation Process

23.1.1 Defining the Structure of the Prepayment Model

The Prepayment Model consists of the Prepayment Dimensions and the Bucket Values for these Dimensions which you select on this page. To define the prepayment model structure, you can select a maximum of three prepayment dimensions. Once the dimensions and the number of buckets (tiers) are defined, you need to assign values to the buckets.

Follow the steps mentioned in Manual section of Prepayment Model Calculation

23.1.2 Prepayment Rate Calculation

Select Manual, Integrated Model, or external models. For Manual mode, You can choose any Prepayment dimension (max. three), define buckets, bucket values and finally explicitly populate prepayment rate matrix. For Integrated Model, prepayment dimension, used in Model would be auto populated as per the selected model. For external mode, you need to define the equation which would later be used for prepayment rate calculations.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Manual
- Integrated Model
- External Model

23.1.2.1 Manual

Using Manual method, you can select maximum of three prepayment dimension and assign prepayment rates manually to selected dimension.

> **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).

Below are the steps for Manual method

1. Select the first Dimension.
2. Select a lookup method for that Dimension.
3. Enter the number of Buckets for the Dimension
4. This number may vary from dimension to dimension.
5. If required, repeat the previous three steps for up to two additional Dimensions.
6. Click **Apply**.
   The Define Dimensions page is refreshed. You can now assign the bucket 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 buckets to a particular Dimension, update the number of buckets for the Dimension and click **Apply**.
   - To delete buckets from a particular Dimension, reduce the number of buckets to the desired value and click **Apply**.

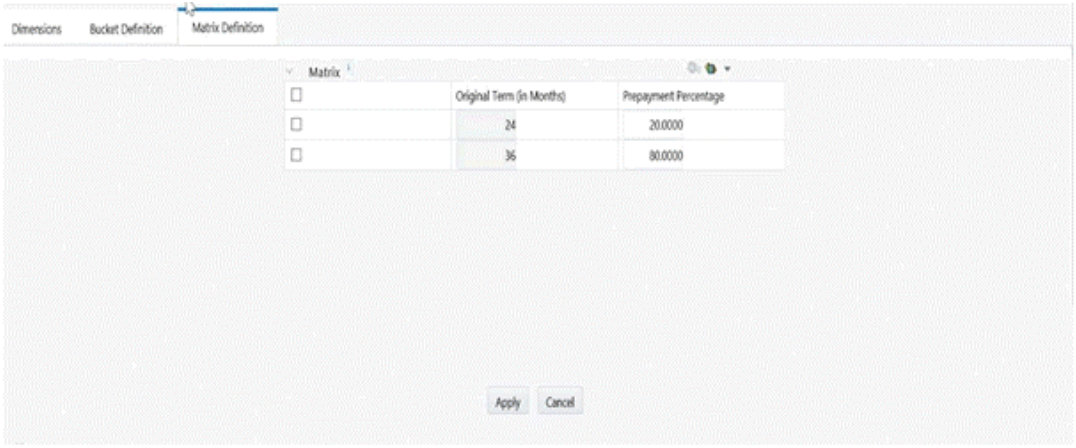
   > **Note**
   >
   > There are certain restrictions while defining Dimensions:
   >
   > You must select the Dimension type for a row and define the values for that dimension.
   >
   > You cannot define the second (row) dimension until you have defined the first (row) dimension. Similarly, the third dimension cannot be defined until you have defined the first two dimensions.

7. Assign values for each of the buckets.
8. Click **Apply**.
   The Prepayment Model, Prepayment Dimensions, and Buckets are saved.
9. Enter the Prepayment Rates in the Prepayment Model.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Bucket values for the row and column dimensions are displayed as a table, while the bucket values for the page dimensions (if selected) are shown in the drop down list.

Figure 23-2 Manual Prepayment Model



Repeat the process for all bucket values of the page driver. To change the bucket value along the page driver, select the required value from the drop-down list.

> **Note**
>
> Bucket values will be displayed in the drop-down list only if you selected three drivers.

10. Click **Apply**. The Prepayment Rates are saved and the Prepayment Model Rule summary page is displayed.

23.1.2.2 Integrated Model

> **Note**
>
> This section is not applicable for CFE.

In Integrated Model option, User has the option to select one of the already saved/confirmed Models. For more information on how Model is defined.

Below steps details how Integrated Model can be used.

1. Select Integrated Model name from Model drop-down list. You can click on Model Search to search an Integrated Model based on certain product and currency combination.
2. Click Apply after selecting Model.
3. Model dimensions would be locked out based on the selected model. Still you need to select lookup method and buckets for each dimension.
4. Define the bucket values for each model dimension.

Copyright © 2011, 2026, Oracle and/or its affiliates.

5. In this mode, prepayment rates would be auto calculated based on the selected model and buckets definition for each dimension. As shown in below screen, click ‘Apply’ to populate prepayment rate matrix. If user would like, one can still edit the values for few cells in prepayment matrix and ‘Save’.

> **Note**
>
> User needs to click on Apply on each page level to save Prepayment rate displayed on that page.

6. In case, model has calculated negative prepayment rate for few cells based on bucket values, you would not be able save the prepayment matrix until all the negative values has been manually updated by the user; as only zero or positive prepayment rates are allowed.
7. A ‘Reset’ button is given, in case user has changed prepayment rates for few cells. You can still get the model calculated prepayment rates back by pressing ‘Reset'.

23.1.2.3 External Model

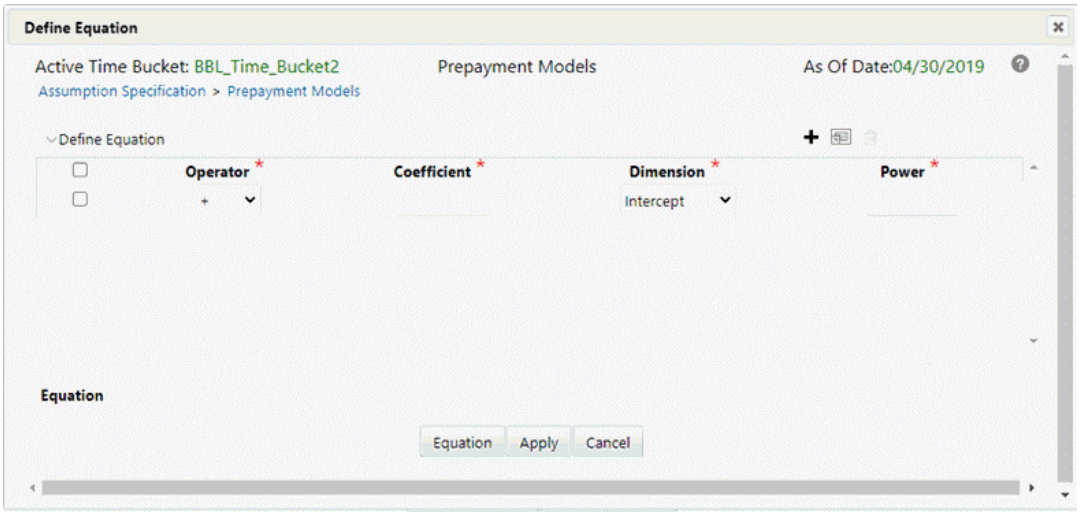
When you select External Model, Define Equation button will get activated to use external prepayment model. This is useful, when you want to do prepayment modelling outside OFSAA and use the model equation to calculate prepayment rates.

1. Define the Dimensions and click Define Equation.

Figure 23-3 Equation

Operator	Coefficient	Dimension	Power
+	2	Intercept	Disabled
+	1.5	Original Term	2
+	3	Rate Diff	2

Figure 23-4 External Model



Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Define the Equation.
   For Example:

   Equation becomes:

   2 + 1.5 * original Term ^ 2 + 3 * Rate Diff ^ 2

> **Note**
>
> Before defining equation, you must select dimensions and accordingly dimensions drop-down will display values along with **Intercept**. For example, if you have already chosen Original term and Rate Difference as dimensions, then Dimension drop-down list would displays the following three values:
>
> • Intercept
>
> • Original Term
>
> • Rate Difference
>
> Following mathematical operators are available: +, -, /and *. After defining all coefficients, Power, operators , click Equation to get the model equation. A confirmation message is displayed. Click Ok to use the same for prepayment rate calculations.
>
> You can add new row for each term using Add +. Multiple rows can be added using Add Multiple Rows

3. Click Apply on Define Equation window. Click Cancel to navigate back to Prepayment Matrix Screen window.
4. Once buckets are defined for final prepayment matrix, matrix would be auto populated.
5. Click Apply to save the final prepayment rates matrix as per defined buckets.
   Note: User needs to click on Apply on each page level to save Prepayment rates displayed on that page.
6. In case, model has calculated negative prepayment rate for few cells based on bucket values, you would not be able save the prepayment matrix until all the negative values has been manually updated by the user; as only zero or positive prepayment rates are allowed.
7. A ‘Reset’ button is given, in case user changed a few cells. You can still get the model calculated prepayment rates back again by pressing ‘Reset’

23.1.3 Editing Prepayment Models

As part of editing Prepayment models, you can modify Prepayment rates and the structure of the Prepayment Model. You can also modify the lookup methods (Range or Interpolation), the number of Buckets, and the actual values of the Buckets. However, if you update the dimensions your previously entered prepayment rate data may be lost.

Prerequisites

Predefined Prepayment Models.

Procedure

1. Search for the Prepayment Model, which you want to edit. For more information, refer to Searching for Rules section.
2. Select the check box next to the Prepayment Model and select the **Edit**

Copyright © 2011, 2026, Oracle and/or its affiliates.

The Prepayment Model opens in Edit mode.

Procedure to Update Rates

Modify the Prepayment Rates in the table as required. For more information, see the Editing Prepayment Rates in a Prepayment Model.

Procedure to Update Dimensions Values

1. Click Update Dimensions Values.
   The Update Dimensions page is displayed.
2. Update the structure as required.
   - 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.
   - To change the value of a Node, update the value in the corresponding field.
3. Click **Apply**.
   The Prepayment Model summary page is displayed.

23.1.4 Editing Prepayment Rates in a Prepayment Model

Once the basic structure of the prepayment model has been created, prepayment rates can be added to, or modified for, each of the bucket values along the chosen dimensions. Use this procedure to add or edit annual prepayment rates in the prepayment model.

Procedure

1. Search for the Prepayment Model, for which you want to define prepayment rates. For more information, see the Searching for Rules section.
2. Select the check box next to the Prepayment Model and select the **Edit** icon.
   The Prepayment rate input page is displayed.
3. Enter the Prepayment Rates in the Prepayment Model for the required dimensions.
   Node values for the row and column dimension are displayed as a table on the Prepayment rate input, 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> Node values will be displayed in the drop-down list only if you selected three dimensions. If prepayment rates are left blank for any of the dimensions, prepayment rates are assumed to be zero.

5. Click **Apply**.
   The table with updated prepayment rates is saved.
6. Click Save.
   All updates to the Prepayment Model are saved and the Prepayment Model summary page is displayed.

Editing Prepayment Rate Calculation

As part of editing Prepayment Rate Calculation, you can change the Prepayment Rate Calculation and the structure of the Prepayment Model. When you change the Prepayment Rate Calculation, a confirmation message is displayed.

Select the Prepayment Rate Calculation model from Prepayment Rate Calculation drop-down list.

- If you select Prepayment Rate Calculation as Manual, you can edit Dimensions, Bucket Definition, and Market Definition.
- If you select Prepayment Rate Calculation as Integrated, follow the instructions mentioned in Using Integrated Model section.
- If you select Prepayment Rate Calculation as External, follow the instructions mentioned in Using External Model section.

Excel Import/Export

Excel import/export functionality is used for adding/editing prepayment rate information. For more details, refer Excel Import/Export.

23.1.5 Editing Prepayment Rate Calculation

As part of editing Prepayment Rate Calculation, you can change the Prepayment Rate Calculation and the structure of the Prepayment Model.

When you change the Prepayment Rate Calculation, a confirmation message is displayed.

The edit the prepayment rate calculation, select the Prepayment Rate Calculation Model from the Prepayment Rate Calculation drop-down list.

- If you select Prepayment Rate Calculation as Manual, you can edit Dimensions, Bucket Definition, and Market Definition.
- If you select Prepayment Rate Calculation as Integrated, follow the instructions mentioned in Using Integrated Model section.
- If you select Prepayment Rate Calculation as External, follow the instructions mentioned in the Using External Model section.

Excel Import or Export

Excel Import or Export functionality is used for adding or editing Prepayment Rate information. For more details, refer to Excel Import or Export.

Copyright © 2011, 2026, Oracle and/or its affiliates.

23.1.6 Prepayment Rate Data Loader

The Prepayment Rate Loader procedure populates prepayment model rates in OFSAA metadata tables from staging tables. This loader program can be used to update the prepayment model rates on a periodic basis. For more information on setting up the automated process, see Data Model Utilities User Guide.

To execute the Data loader process, click the Data Loader icon on the Prepayment Models rule bar. The process will load all of the data included in the staging tables as defined in the Data Model Utilities User Guide

23.1.7 Prepayment Model Analysis

This module describes the procedure for prepayment models analysis.

The Prepayment Model Analysis module will allow you to develop a model for prepayment rate calculations as per the chosen risk factors.

Model building could be a time consuming process given huge data sets and limited IT infrastructure. So to get the latest status of each stage, you can hover over corresponding block in the progress train, a call out will open with milestones status which will help to confirm if the process is still running and how many milestones have already completed.

> **Note**
>
> Before using the product hierarchy for selecting data to include in your process, there is a procedure that must be run (`PRODUCT TO INSTRUMENT MAPPING`). This procedure can be executed from the Batch Scheduler – Run - interface.

The purpose of the Product to Instrument mapping procedure is to scan all instrument tables (FSI_D_xxx) and populate the mapping table (`FSI_M_PROD_INST_TABLE_MAP`) with a listing of the product dimension members that exist within each instrument table.

> **Note**
>
> Prepayment model analysis is supported only for OEL.
>
> For User Roles and Functions related to Prepayment Model Analysis, see the Creating Application Users section of OFS Asset Liability Management Application Pack Installation and Configuration Guide.

In the Oracle Financial Services Analytical Applications Infrastructure home screen, select **Financial Services Asset Liability Management**.

Note: For FTP, select **Financial Services Funds Transfer Pricing Management**. For BSP, select **Financial Services Balance Sheet Planning Management**

1. To access Prepayment Model Analysis window, navigate to ALM Assumption Specification, select **Prepayment Rules** and then select **Prepayment Model Analysis**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> For FTP, navigate to ALM Assumption Specification. For BSP, navigate to BSP Assumption Specification.

2. The definitions based on the search criteria are listed under the List of Prepayment Model Analysis. You can search on one or more model attributes in this section. When you click Search, depending upon the search criteria, the corresponding Prepayment Model Analysis Summary would be shown. The Prepayment Model Analysis Summary window displays the following fields.

Table 23-2 Fields and their Descriptions

Fields	Description
Search	This icon allows you to search the Prepayment Model Analysis definition based on the search criteria specified. Search criteria may include a combination of Name, Folder, Status, and Model Type. The Prepayment Model Analysis rule definitions displayed in the Prepayment Model Analysis Summary table are filtered based on the search criteria specified on clicking this icon.
Reset	This icon allows you to reset the Search section to its default state without any selections. Resetting the Search section displays all the existing Prepayment Model Analysis rule definitions in the Prepayment Model Analysis Summary table.
Name	This field allows you to search the pre-defined Prepayment Model Analysis rule based on the Rule name. Enter the Rule name.
Folder	This field allows you to search for the pre-defined Prepayment Model Analysis rule definitions based on the selected folder. This field displays a list of folders that you have access to as a drop-down. Selecting a folder from the drop-down list displays only those Prepayment Model Analysis rule definitions that belongs to the selected folder/segment in the Prepayment Model Analysis rule Summary table.
Model Type	This field allows you to search the pre-defined Prepayment Model Analysis rule based on type. Model Type would only be defined for “Confirmed” models. All models which are still in draft, would not have any Model type. This drop-down has the following options: • Linear • Polynomial
Status	This field allows you to search the pre-defined Prepayment Model Analysis rule based on status. This drop-down list displays the statuses Draft or Completed. Selecting a status from the drop-down list displays only those Prepayment Model Analysis rule definitions that are defined with the selected status in the Prepayment Model Analysis Summary table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The **Prepayment Model Analysis** Summary page offers several icons that allow you to perform different functions when a Prepayment Model Analysis rule is selected.

Table 23-3 Prepayment Model Analysis rule – Icons and Descriptions

Fields	Description
Add	Click Add icon to build a new Prepayment Model Analysis rule. The Add icon is disabled if any row is selected in the table.
View	Select a single row in the table to enable the View icon. Click View to view the contents of a Prepayment Model Analysis rule in read-only format. The View icon is enabled only when a single Prepayment Model Analysis rule is selected in the table.
Edit	Selecting a single row out of the pane enables the Edit icon. Click the Edit icon to modify a previously saved Prepayment Model Analysis rule. The Edit icon is only enabled when a single Prepayment Model Analysis rule is selected. You can control the number of rows to display on the window by selecting the Pagination Options icon from the action bar.
Copy	Selecting a single row out of the pane enables the Copy icon. Click the Copy icon to create a copy of an existing Prepayment Model Analysis rule. The Copy icon is only enabled when a single Prepayment Model Analysis rule has been selected.
Check Dependencies	Select the Prepayment Model Analysis rule. Click the Check Dependencies to generate a report on all rules that utilize your selected Prepayment Model Analysis rule. The Check Dependencies icon is only enabled when a single Prepayment Model Analysis rule has been selected.
Delete	Select one or more rows in the table to enable the Delete icon. Click Delete to delete the selected Prepayment Model Analysis rule. Only prepayment models that are not being used, could be deleted.
Refresh	Click Refresh to refresh the Prepayment Model Analysis summary page.

To select a Prepayment Model Analysis, select a check box in the first column of the table. More than one Prepayment Model Analysis can be selected at a time, but this disables some of the icons.

You can select or deselect all the Prepayment Model Analysis rules on the Summary page by selecting the check box in the upper left-hand corner of the Summary table directly to the left of the Name column header.

The following table provides the Prepayment Model Analysis details based on the search criteria.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 23-4 Prepayment Model Analysis – Columns and Descriptions

Column	Description
Name	Displays the Prepayment Model Analysis rule's short name.
Creation Date	Displays the Date and Time at which a Prepayment Model Analysis rule was created.
Created By	Displays the Name of the user who created the Prepayment Model Analysis rule.
Last Modification Date	Displays the Date and Time at when a Prepayment Model Analysis rule was last modified.
Last Modified By	Displays the Name of the user who last modified the Prepayment Model Analysis rule.
Status	Displays the Status of rule.

Topics:

- Searching for a Model Rule
- Defining a Model
- Editing Prepayment Model Analysis Rule
- Deleting Prepayment Model Analysis Rule
- Copying Prepayment Model Analysis Rule
- Dependency Checking

- Searching for a Model Rule
- Defining a Model
- Editing Prepayment Model Analysis Rule
- Deleting Prepayment Model Analysis Rule
- Copying Prepayment Model Analysis Rule
- Dependency Checking

23.1.7.1 Searching for a Model Rule

Search for a Model to perform any of the following tasks:

- Update, Copy or Delete existing rules.
- Define methodologies for products or define other processing assumptions

To search for a model, follow these steps:

1. Navigate to the **Prepayment Model Analysis** window.
2. Enter the following details:
   - Name
   - Folder
   - Model Type
   - Status
3. Click **Search** .

Copyright © 2011, 2026, Oracle and/or its affiliates.

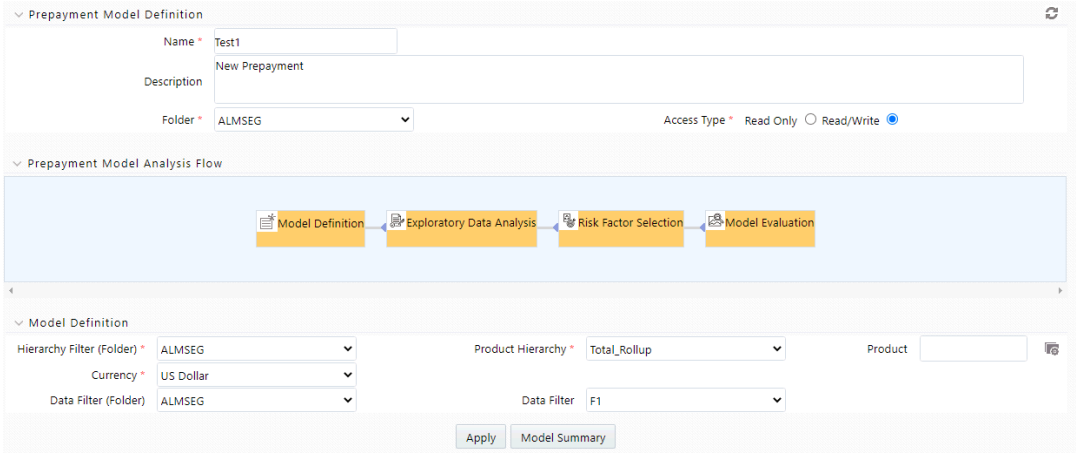
Only rules that match the search criteria are displayed.

23.1.7.2 Defining a Model

To define a model, follow these steps:

1. Navigate to **Prepayment Model Analysis** window. Click **Add** . The Definition window is displayed.

Figure 23-5 Prepayment Model Analysis



2. Enter the following details.

Table 23-5 Form Fields to define a Model

Fields	Description
Name	Enter the name of Prepayment Model Analysis rule
Description	Enter the description of Prepayment Model Analysis rule
Folder	Select the folder where you want to save Prepayment Model Analysis rule
Access Type	Select Access Type as Read-only or Read/Write.
Hierarchy Filter (Folder)	Select the folder of Hierarchy Filter.
Product Hierarchy	Select the Product Hierarchy.
Product	Based on the selected product hierarchy in model definition UI, Hierarchy Browser will expand the product hierarchy and you can select any parent/leaf node by clicking on the view button next to the product in the model definition screen.
Currency	Select the currency for the Prepayment Model Analysis rule. Default Currency option is also given, if you want to select the product irrespective of currency in which deal is booked.
Data Filter (Folder)	Select the folder name of data filter for Prepayment Model Analysis rule

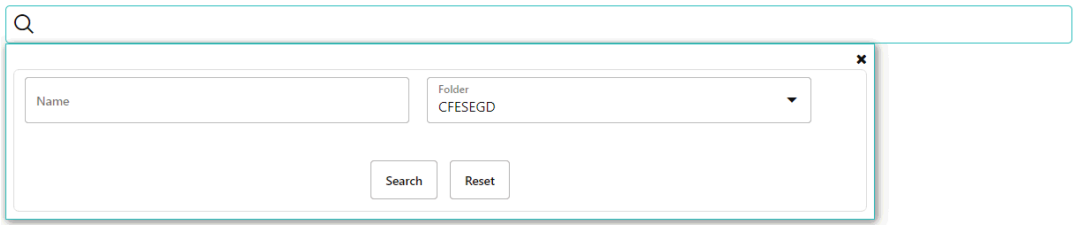
Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 23-5 (Cont.) Form Fields to define a Model

Fields	Description
Data Filter	Select the data filter to define the portfolio at more granular level than just, product and currency combination.

3. Click **Apply** to navigate to **Exploratory Data Analysis** window.
4. Exploratory Data Analysis window is used to perform EDA calculations.

Figure 23-6 Exploratory Data Analysis



Enter the following details in Exploratory Data Analysis window.

Table 23-6 Form Fields of Exploratory Data Analysis window

Fields	Description
Sample Size	Sample size of dataset for EDA
Model Population (As of Date Range)	You can define a date range within which one wants to use data for model creation as model population. By Default, Date range would be populated as below: End Date: Max (As of Date) available in Model input Table Start Date: Max [min (As of Date), End Date – 10 Years]
Do Sampling	Select this option to create model based on a data Sample rather than the whole population. By default, Do Sampling is enabled. This would enhance performance due to lesser number of records considered for modelling, without degrading model quality.
Sample Size (Multiplier)	After selecting sample size for EDA, you can select multiplier value. This indicates sample size required for model creation. For example, if 1000, records are selected as sample size for EDA, and multiplier is 6, then a minimum of 1000*6 = 6000 records would be required for model creation. Note: This checkbox is enabled only if Do Sampling checkbox is selected.

5. This window gives complete information about all the risk factors along with prepayment rates. It helps you do decide what all factors could influence the customer's prepayment behavior and would be best for model building.

Copyright © 2011, 2026, Oracle and/or its affiliates.

You can hover over any graph and zoom in to enhance the visibility of the graph.

Click Model Summary to navigate back to the Model Summary window after saving all inputs/EDA graphs defined/generated till this point.

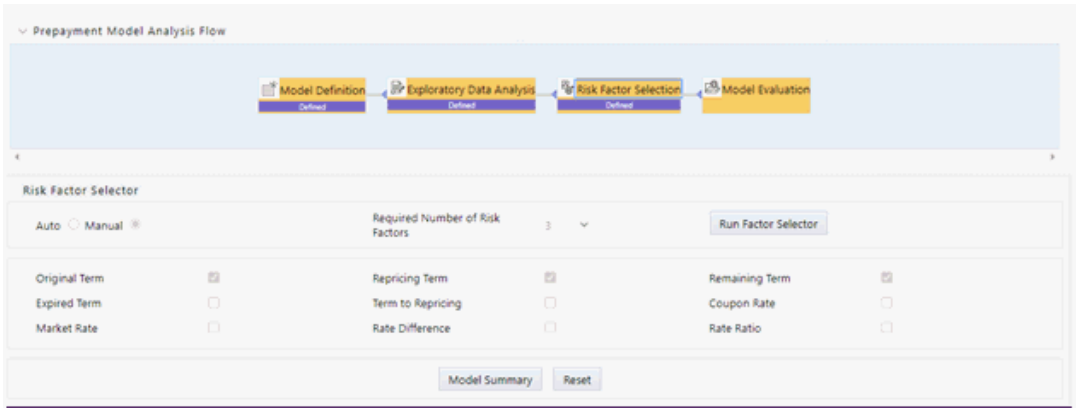
Click Re-Calculate if you want to change the sample size and redo the EDA again. change the sample size and click Re-Calculate.

If you have performed EDA multiple times (for example, three versions V1, V2, V3.), then use Versions (EDA) to view them. For example, when you are on the EDA screen and performing the EDA 3rd time, but still want to go with the 2nd EDA version, then select that version and subsequent processing would be based on V2. If you have run the EDA only once, then this drop-down will not be available.

When you hover over Sample size for EDA, it displays Default value (callout) as Default Value is 5000.

6. Click Pair Plot to generate Pair Plot/Grid along with other EDA graphs. Pair Plot/Grid is a detailed graph, that can further slow down the processing. So you can explicitly select the pair plot checkbox and click on calculate to perform the EDA.
7. Click Next. The **Risk Factor Selection** window is displayed. By default, all the risk factors will be disabled.

Figure 23-7 Risk Factor Selection window

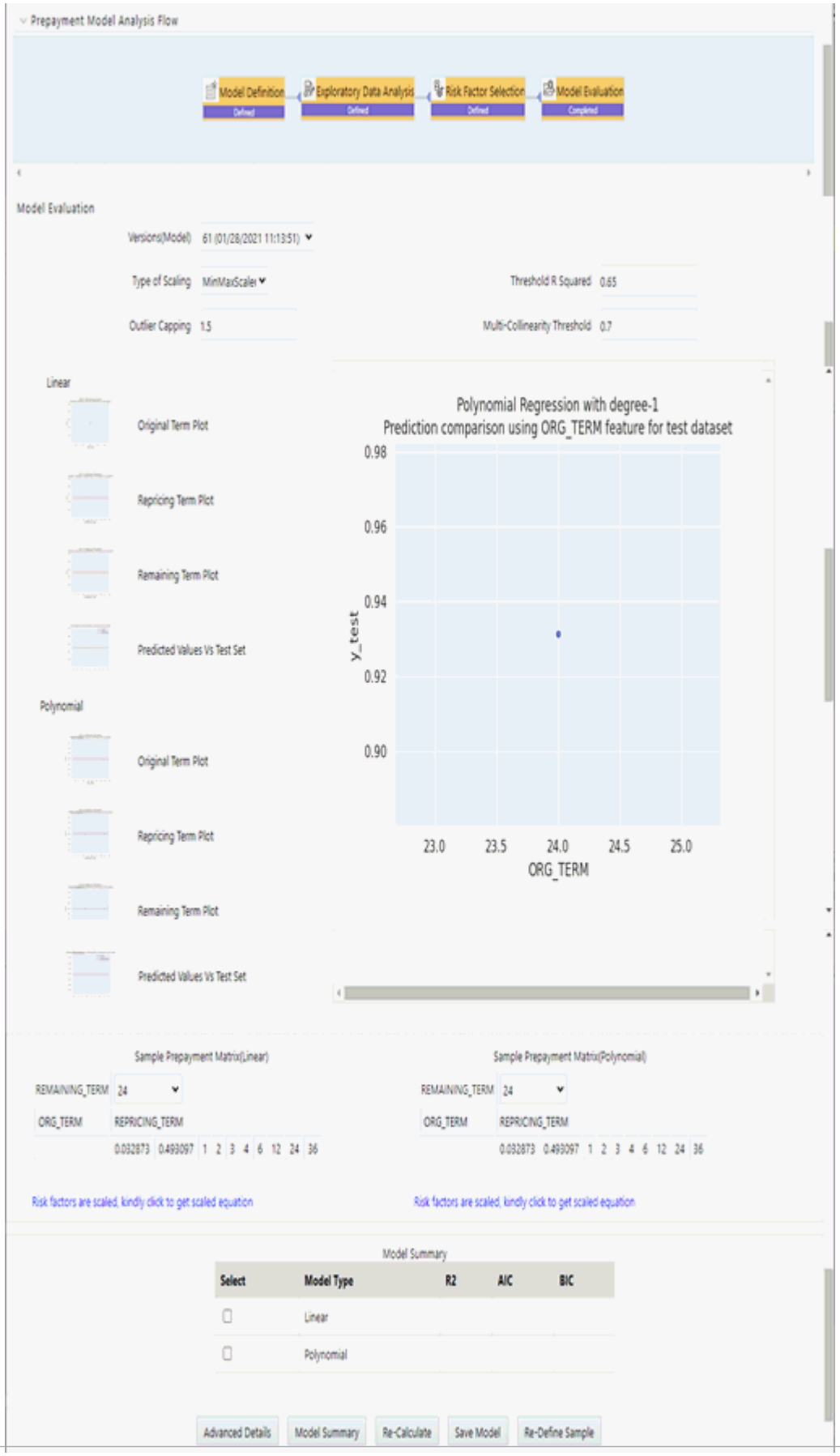


8. You can change the risk factor selection mode to ‘Manual’ and all the risk factors would be available for selection; maximum 3 factors can be selected after that all the risk factors would be disabled again.
When you are in Auto Select Mode, System would perform required calculations, co-relation/collinearity analysis in the backend based on the required number of risk factors (Maximum 3), System would auto-select the best representative set of risk factors as per the input data.

   If you want to change sample size and re-calculate EDA again, click Exploratory Data Analysis block and perform EDA with updated sample size. Again, the process starts from that step onwards, and updated EDA plots/graphs would be saved for the model.
9. Click Next to navigate to **Model Evaluation** window. A confirmation message is displayed. Click OK.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 23-8 Model Evaluation



Copyright © 2011, 2026, Oracle and/or its affiliates.

10. Click Calculate to view all the evaluation parameters and quality plots.
11. Click Re-Calculate if you are not satisfied with the model quality to change the model parameters and revise the model definition. You can zoom the graph to enhance visibility. If you are changing parameters and generating different versions of the model, then you can see all the versions, generated with different set of model parameters in the Versions (Model) drop-down. If you have run “model evaluation’ only once, then this drop-down would not be available. As per the selected version, the Advanced model details window is also updated. The Model Details window helps you to evaluate the model fit. It has the following comparisons:

    Predicted Values Vs Risk Factor 1,2,3, as per the number of risk factors considered for model building. This will dynamically adjust as per the number of risk factors selected.

    Predicted Values Vs Test Sample.

    Sample prepayment matrix as per both the models (Linear and Polynomial).

Figure 23-9 Sample prepayment matrix

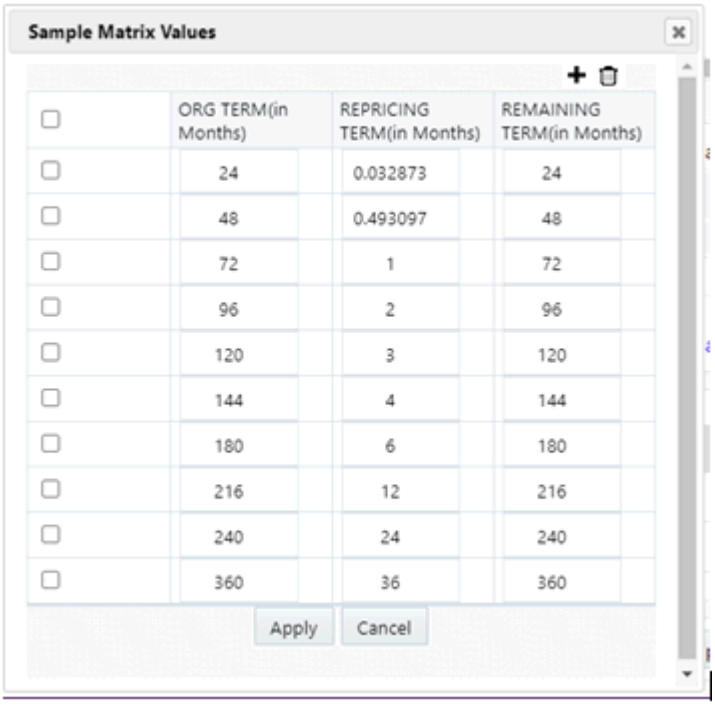


    User would have option to redefine, values for sample matrix by clicking on “Re-define Sample” button. Below pop up would open up if user would like to re-define sample buckets. Again this pop up would dynamically adjust based on number of risk factors selected in the model.

12. Click **Model Summary** to view Model Summary window, after saving all details.
13. Click **Re-Define Sample** if you want to modify risk factor values for sample matrix. This pop up would dynamically adjust to display 1,2 or 3 dimensions as per the chosen model.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 23-10 Risk factor values



In case you want to update values on which sample matrix is generated, you can click on 'Re-define Sample'. Below window would open up where you can update the risk factor values:

Add row using **Add** +. You can add multiple rows or delete multiple rows using buttons in panel 1. Default sample matrix is10*10, but once you re-define sample, matrix can be truncated to any dimensions.

The redefined matrix will be saved along with model.

Model summary parameters like R2, AIC and BIC.

Both (Linear/Polynomial) models are produced when the system compares model generated R2 against the R2 threshold defined by the user. The final option is given to you to choose any one of these models. Based on infrastructure availability and model complexity, user can choose any one of linear/polynomial model.

Click **Save Model** to save the model. Same model could be referred to populate prepayment rate matrix.

Click Re-Calculate to re-evaluate model based on changed model parameters.

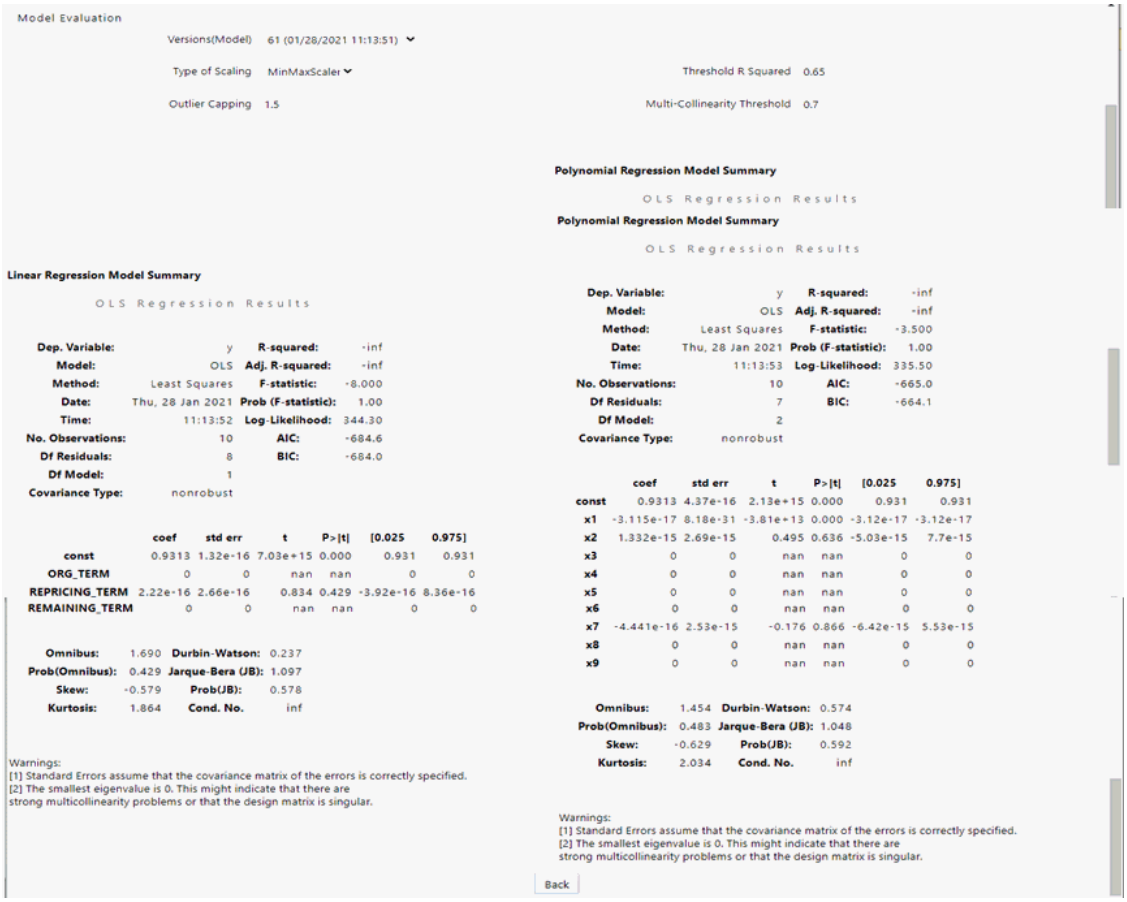
If you want to change sample size and re-calculate EDA again, click Exploratory Data Analysis block and perform EDA with updated sample size. Again, process would start from that step onwards and updated EDA plots/graphs would be saved for the model.

If you want to update selected risk factors, click Risk Factor Selection block and change risk factors. Again, process would start from that step onwards and updated risk factors would be saved for the model.

Advance Details – To verify the model quality, all the model statistics are given on a different screen, which would be available with 'Advance Details' button like R2, F value, P – Value, and so on.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 23-11 Advance Details



Reset button on each screen would help to delete all the calculations done in subsequent steps. That is, if you have done EDA and selected particular EDA version to do further calculations like ‘Risk factor Selection’ or ‘Model Evaluation’. In case you are not satisfied with model, you can go back to EDA, click ‘Reset’ to clear out the details/calculations performed in subsequent stages and do recalculate with a different set of parameters.

Following are the default values/usage of Parameters:

Default values/usage of Parameters

Parameters	Default Values
EDA Sample Size	This would allow you to define a sample size for exploratory data analysis. A bigger sample would increase CPU and memory usage, but it would better represent the model population. You have the option not to use sampling by setting Do_Sampling parameter to false. Procedure for the same is given in next section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Parameters	Default Values
Type of scaling	Many a times, risk factors are not in consistent range, e.g. one of the risk factor's values could be in 1-500 range but another risk factor could be just in 2-3 range. So, risk factor 1 would influence the model more and you would get a biased model. So, to make all the risk factors consistent, scaling is used. There are two types of scaling: Min-Max Scaling = (X – min)/(max – min) Standard = (X – Mean)/Std. Deviation
Threshold R2	R-squared values range from 0 to 1 and are commonly stated as percentages from 0% to 100%. An R-squared of 100% means that all movements of prepayment rate (dependent variable) are completely explained by movements in the chosen risk factors (independent variable(s).
Multi-collinearity Threshold	For model creation if two risk factors/variables are highly correlated or correlation > 0.7, they would make model unstable. So based on this value, if variables are highly correlated or above defined threshold, one of them would be dropped while model creation.
Outlier Capping	This would allow user to reject values beyond certain percentile. Sometimes, input data has few extreme values which could distort the model. So you could reject those values and get a stable model.

Exploratory Data Analysis:

Sample Size (EDA) - 5000

Model Population Range - It would be auto-populated based on the data in risk factor table. Maximum would be 10 years older from latest available date. In case you think, older data is not relevant, as of date range can be updated.

Model Evaluation:

Type of scaling – min Max Scaler

Threshold R2 – 0.65

Outlier capping – 1.5 Percentile

Multi-collinearity Threshold – 0.7

23.1.7.3 Editing Prepayment Model Analysis Rule

To edit a rule, perform following steps:

1. Navigate to **Prepayment Model Analysis** window.
2. Search for Prepayment Model Analysis rule which you want to edit. For more information, see the Searching for Rules section.
3. Select the Prepayment Model Analysis rule. Click Edit to open the rule you want to update. You cannot edit more than one Prepayment Model Analysis rule at a time. The Edit Definition window is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

4. Modify the details, only model type can be changed from linear to polynomial or vice versa and click Apply.
5. Click Save to navigate back to Prepayment Model Analysis summary window.

23.1.7.4 Deleting Prepayment Model Analysis Rule

You can delete rules that are no longer required.

> **Note**
>
> A rule cannot be retrieved after deletion.

Restrictions on deleting rules are:

- You cannot delete rules if you have only Read privileges. Only users with read/write privilege and rule owners can delete models.

- You cannot delete a rule that has a dependency, that is, model is being used in any of the prepayment matrix to calculate prepayment rates.

To delete a rule, follow these steps:

1. Navigate to the **Prepayment Model Analysis** window.
2. Search for a Prepayment Model Analysis rule. For further information, see the Searching for Rules section.
3. Select the Prepayment Model Analysis appropriate rule and click **Delete**.

23.1.7.5 Copying Prepayment Model Analysis Rule

You can copy rules to avoid having to enter data multiple times. This saves time and effort and also reduces mistakes.

To copy a rule, follow these steps:

1. Navigate to the **Prepayment Model Analysis** window.
2. Search for a Prepayment Model Analysis Rule. For further information, see the Searching for Rules section.
3. Select the appropriate Prepayment Model Analysis rule and click **Copy** to duplicate the rule.
4. Select a folder where you want to save the Prepayment Model Analysis rule copy.
5. Enter a unique name for the new Prepayment Model Analysis rule.
6. (Optional) Enter a brief description of the rule.
7. Select the access type.
8. Click **Save** .

23.1.7.6 Dependency Checking

You can check dependencies for rules to know where a particular rule, dimension, member, attribute, IRC has been used. This also prevents accidental deletion of rules having dependencies.

Copyright © 2011, 2026, Oracle and/or its affiliates.

To check the dependency of a Prepayment Model Analysis rule, follow these steps:

1. Navigate to the **Prepayment Model Analysis** window.
2. Search for a Prepayment Model Analysis rule. For further information, see the Searching for Rules section.
3. Select the appropriate Prepayment Model Analysis rule or rules and click **Check Dependencies** corresponding to the rule that you want to check for.

> **Note**
>
> You can select more than one rule at a time to check dependencies.
>
> The Dependency Information window opens containing the following information: Child Object Name, Child Object Type, Folder, Parent Object Name, Parent Object Type, and Folder.
>
> You cannot delete a child object which has dependencies. You will get a message: *Dependencies found. Cannot delete*.
>
> To delete the child, you must first delete the outermost parent object which uses the child.

23.1.8 Prepayment Rate Calculation Process

Prepayment are associated with Assets and Liabilities that have fixed maturities and have experienced a change in scheduled contractual cash flows on a fixed maturity instrument results in a prepaid event and should be captured.

The Prepayment Rate Calculation Process helps you to create data for the subsequent step prepayment modelling by performing following tasks:

- Define the product for which prepayments behavior need to be learnt (Product Selection block).
- Specify the parameters to calculate various risk factors which can influence prepayments, such as, As of Dates reference term, threshold prepaid amount, Market rate calculation parameters.
- Execute or run the Prepayment Rate Calculation process and generate results through Prepayment Rate Calculation Process Summary Page.

Topics:

- Creating a Prepayment Rate Calculation Process
- Executing a Prepayment Rate Calculation Process
- Viewing Prepayment Rate Calculation Process
- Editing Prepayment Rate Calculation Process
- Deleting Prepayment Rate Calculation Process
- Performance Tuning Prepayment Rate Calculation Processes
- Creating a Prepayment Rate Calculation Process
- Editing an Existing Prepayment Rate Calculation Process
  This section details the procedures for viewing, editing, copying, and deleting and existing Prepayment Rate Calculation Process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Executing a Prepayment Rate Calculation Process
- Performance Tuning Prepayment Rate Calculation Processes

23.1.8.1 Creating a Prepayment Rate Calculation Process

The Prepayment Rate Calculation Process will compare current period instrument data with prior period based on the ID number to identify prepaid events. When prepayments are detected, the related record's prepayment rate and following risk factors:

- Original Term
- Repricing Term
- Remaining Term
- Expired Term
- Term to Reprice
- Coupon Rate
- Market Rate
- Rate Difference
- Rate Ratio

This is copied to the FSI_O_HIST_PPMT_RATE_DETAILS table. Later, this table will be used as source for prepayment rates modelling.

To create a Prepayment Calculation Process, follow these steps:

1. From the LHS menu, select Asset Liability Management, select ALM Processing, and then select Prepayment Rate Calculation Process to display the Prepayment Rate Calculation Process screen.
   The following table describes key terms used for this procedure.

Table 23-7 Fields for creating a Prepayment Rate Calculation Process

Term	Description
Folder	The folder where you can save the definition. You can give other users, read/write or read only privileges.
Filter	Filters allow you to restrict your data selection based on any various account attributes like fixed/floating rate, amortization code etc. You can define filters under Common Object Maintenance and reference your filter within the Product Selection block of your Process. The choice of the data filter would determine the instrument records that should be picked from current and prior As-of-date for comparative analysis.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 23-7 (Cont.) Fields for creating a Prepayment Rate Calculation Process

Term	Description
Source	Allows you to select one or more source Instrument tables to include in your process. Based on the Instrument Table(s) selected, the instrument records from current and prior As-of-date date would be picked for comparison. Following table are available as source: Loan Contracts Mortgages Term Deposits
Minimum Prepaid Amount	Minimum Prepaid amount applies to both Positive and Negative prepaid amounts. If the user enters minimum prepaid as 50, it means that minimum prepaid amount ranges from -50 to +50. If the Prepaid Amount falls within this range, then the record would not be considered for any calculations.
Freeze Process	The freeze button will help you to confirm if the process is complete and ready to run or further changes are required and it is an incomplete process which can not run.

2. Product Selection: This block will help you to define the accounts where prepayments took places and should be considered to learn prepayment behavior.
3. Product Selection Block:

§ Select a Filter (optional) to constrain the data to be included in the process.

§ Select the source table(s) that you want to include in the process.

4. Parameters:

§ Input the Minimum Prepaid Amount as a positive value. The engine applies the absolute value of the amount input ranging from - input amount to + input amount. For example, if the input is 100, then prepaid amounts between -100 and +100 is excluded. This input allows you to filter very small / insignificant prepaid amounts, reducing the amount of data copied into risk factor table for further modelling process.

§ Note there are two approaches for determining the Prior Period Date. You can input the Prior Period Reference Term and based on the current As-of-Date set in application preferences, the Prior Period Date will be calculated, or you can select the “Use Nearest Prior Date” option, and the engine looks back at the historical data to determine the nearest prior As-of-Date and uses this as Prior Period Date.

§ One of the risk factors is customer rate. It is generally picked from CUR_NET_RATE, but f user has specifically chosen gross rate from UI by checking ‘Model with Gross Rate”, CUR_GROSS_RATE is picked from the instrument record.

5. Market Rate Calculation Parameters:

You must select an IRC from the list of IRCs defined in Rate Management > Interest Rates. The selected IRC (Index in Parameter screen) provides the base value for the market rate.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Spread can be positive as well as negative values and both are to be added to base market rate. Therefore, if the base market rate is 5 and spread given is 1.2, the final rate is be 5+(1.2) = 6.2

Similarly, if spread is -1.2, then final market rate is 5+(-1.2) = 3.8

Additionally, you must specify the reference term you want to use for IRCs that are yield curves.

The following are the options for you to select:

§ Original Term: The calculation retrieves the interest rate from the term point equaling the original term on the instrument.

§ Reprice Frequency: The calculation retrieves the interest rate from the term point equaling the reprice frequency of the instrument. If the instrument is fixed rate and, therefore, does not have a reprice frequency, the calculation retrieves the interest rate associated with the term point equaling the original term on the instrument.

§ Remaining Term: The calculation retrieves the interest rate from the term point equaling the remaining term of the instrument.

Effective Date for Market rate Curve:

§ Select the market rate as per the given curve for prior As-of-Date.

§ Select the market rate with some lag say 15 days, 30 days. Lag Term can be defined with a dropdown containing days, months and years.

§ Similarly, market rate can be defined as arithmetic average over historical range of 1 month, 6 months starting prior As-of-Date. Historical term is also given with a drop-down containing days, months and years.

6. Freeze Process:

7. Select Freeze to complete the process.

8. Select Reset to erase all selections made previously within the process definition flow.

9. Select Confirm.

23.1.8.2 Editing an Existing Prepayment Rate Calculation Process

This section details the procedures for viewing, editing, copying, and deleting and existing Prepayment Rate Calculation Process.

- Viewing Prepayment Rate Calculation Process
- Editing Prepayment Rate Calculation Process
- Deleting Prepayment Rate Calculation Process

23.1.8.2.1 Viewing Prepayment Rate Calculation Process

To view any of the saved processes, you can select the process using check box next to each process and click on view button.

23.1.8.2.2 Editing Prepayment Rate Calculation Process

To edit any of the saved processes parameters, you can select the process using check box next to each process and click on edit button

Copyright © 2011, 2026, Oracle and/or its affiliates.

23.1.8.2.3 Deleting Prepayment Rate Calculation Process

To delete any of the saved processes, you can select the process using check box next to each process and click on delete button.

23.1.8.3 Executing a Prepayment Rate Calculation Process

You execute a Prepayment Rate Calculation to compare current period and prior period data to identify different prepayment events.

Performing basic steps for creating or editing a Prepayment Rate Calculation Process is the prerequisite for this.

Following two approaches are available to execute the process:

- UI Based Procedure
- Batch Based Procedure
- UI Based Procedure
- Batch Based Procedure

23.1.8.3.1 UI Based Procedure

To execute the process using UI based procedure, follow these steps:

1. Navigate to the Prepayment Rate Calculation Process summary page.
2. Select a Process that you want to execute or run.

> **Note**
>
> The status column indicates whether or not a process can be run. The following are the possible status conditions:

   - Complete: Indicates the process is fully defined and ready to be run.
   - Incomplete: Indicates the process is partially defined and cannot be run.
   - View Log: Indicates the process has already been run and also indicates a process can be re-run. In this case View Log is a hyperlink that allows you to navigate to the Log Viewer page where you can review any processing errors or alerts related to this process.
3. After executing the preceding process, select the View Log Hyperlink.
4. Select the Task ID (also known as the unique system identifier) to view a report for any processing errors.
   If significant processing errors exist, you should re-run your process.
5. The Prepayment Rate Calculation process is complete.

23.1.8.3.2 Batch Based Procedure

Prepayment Rate Calculation Processes can also be executed as batch processes. The following approaches are available:

- Simplified Batch

Copyright © 2011, 2026, Oracle and/or its affiliates.

- ICC Batch
- Run Rule Framework
- Simplified Batch
- ICC Batch
- Run Rule Framework

23.1.8.3.2.1 Simplified Batch

To execute the batch with Simplified Batch, follow these steps:

1. From the LHS menu, select Common Object Maintenance, select Administration, and then select Simplified Batch.
2. Create a New Simplified Batch.
3. Select Task Type as Transform Data.
4. Select Historical Prepayment Batch from the list of seeded process types.
5. Input the following in the Optional Parameters text box:`'Prepaid Detection Process System ID','User ID', for example, '200883','TPUSER'`
6. Save the rule and Execute.

23.1.8.3.2.2 ICC Batch

To execute the batch with ICC Batch, follow these steps:

1. From the LHS menu, select Common Object Maintenance , select Operations, and then select Batch Maintenance.
2. Create a new batch.
3. Select the Batch Name to add the task.
4. Click the Add button under the Task Details section.
5. Define the Task ID and Description.
6. Select Components as Transform Data.
7. Select Historical Prepayment Batch from the list of seeded process types.
8. Input the following required parameters:
   - Prepaid Detection Process System ID
   - User Name

   These parameters must be given in the follow format (sys Id, User Id), for example: 101000, TPUSER.
9. Save the rule and Execute.

23.1.8.3.2.3 Run Rule Framework

To execute the batch with Run Rule Framework, follow these steps:

1. From the LHS menu, select Common Object Maintenance, select Manage Runs, and then select Process.
2. Create a new rule.
3. Define the Folder, Code, and Name of rule.

Copyright © 2011, 2026, Oracle and/or its affiliates.

4. Select the Component as Transform Data.
5. Select Historical Prepayment Batch from the list of seeded process types.
6. Input the following required parameters:
   - Prepaid Detection Process System ID
   - User Name

   These parameters must be given in the follow format (sys Id, User Id), for example: 101000, TPUSER.
7. Save the rule and Execute.

23.1.8.4 Performance Tuning Prepayment Rate Calculation Processes

Processing time for Prepayment Rate Calculation Processes can be reduced by modifying parallel processing parameters in the FSI_PARALLEL_PARAMETERS table.

Parallel execution related parameters for Prepayment Rate Calculation are seeded by the application with a process_engine_cd value of "3" in the FSI_PARALLEL_PARAMETER table.

The Prepayment Rate Calculation process looks for the value given in parallel_query field. If the value is given as "ENABLED", the engine uses the PARALLEL_DEGREE input and sets the degree of parallelism for execution of the queries within the Prepayment Rate Calculation engine.

These values must be set by the user or implementation team, when they want to take advantage of parallel query executions. The optimal value for the PARALLEL_QUERY input depends on the environment and must be iteratively tested to arrive at the appropriate value.

Copyright © 2011, 2026, Oracle and/or its affiliates.

24

Non-Maturity Products Data Creation Process

To understand the end-user behavior and use the funds available from the Non-Maturity Products portfolio more precisely; in-built PYTHON Models are given. These PYTHON Models need to run on Historical Data like Market Rate, Cur Net rate, Cur Par Bal, and Account Origination Date, to create this data, the Non-Maturity Products Data Creation Process can be leveraged.

Non-Maturity Modeling is comprised of three types of models:

- Core/Volatile Balance segregation
- Decay Rate and corresponding Weighted Average Life Calculation
- Beta Factor or Pass-through Rate Calculations
- For these calculations, the following Portfolio Level Attributes are required:

Instruments in Scope are CASA, Credit Cards, and Ledger Instruments.

- Account Origination Date (To Calculate Vintage/account age)
- Deposit Rate or Bank Rate – CUR_NET_RATE
- Underlying Index (Market Rate) or Reference Rate
- Number of Accounts per vintage
- Portfolio Balance snapped at regular intervals for a given historical period. (CUR_PAR_BAL from each account)

The Data Creation Process allows you to perform the following tasks:

- Defining the Portfolio and the corresponding Historical Period over which data needs to be generated. Select any number of products from the selected Hierarchy and one or multiple currencies to define a portfolio.
- Hierarchy can be defined on any of the product dimensions like Chart of Account, General Ledger accounts, or Product IDs. The system is built in such a way that it can display Hierarchies based on selected Product Dimensions in Application Preferences.
- Process Train is given on each page, so you can keep track of how many stages are defined and how many are yet to be defined.
- Specify the Market Rate Definition and other Process Parameters like Yield Curve Term and Interest Rate Curve Effective Date.
- Review the entered details and all the process parameters.
- Execute the Data Creation Process and generate results.
- Data Generated by Batch UI using following tables:
  - FSI_AUDIT_NM_LOAD_BATCH;
  - FSI_O_HIST_NM_MODEL_DETAILS;
  - FSI_O_IRC_RATE_NM_MODEL;
  - FSI_PROCESS_RUN_HISTORY;
- Metadata tables used for NMD Model UI:

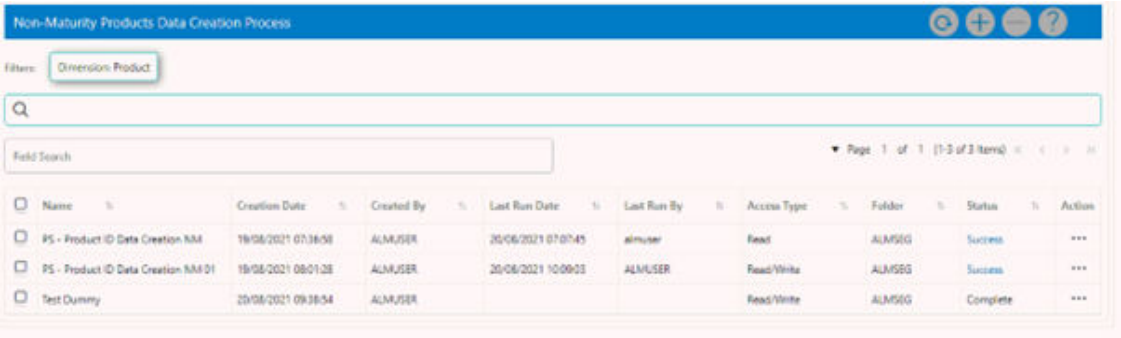
Copyright © 2011, 2026, Oracle and/or its affiliates.

- FSI_M_NM_MODEL_DEFINITION;
- FSI_M_NM_MODEL_EDA;
- FSI_M_NM_PASS_THROUGH_RATE;
- FSI_M_NM_STABLE_BALANCE;
- FSI_M_NM_DECAY_RATE;
- FSI_M_NM_REVIEW_MODEL;

The Data Creation Process summary page allows you to do the following:

- Create a new process.
- Searching for Data Creation processes: For more information, see the Search for Rules Section.
- Creating Prepayment Models: For more information, see the Create Rules Section.
- Viewing and updating Data Creation processes: For more information, see the View and Edit Rules Section.
- Copying Data Creation processes: For more information, see the Copy Rules Section.
- Deleting Data Creation processes: For more information, see the Delete Rules Section.
- Execute the process.
- View the Execution Details.

Figure 24-1 Data Creation Process summary page



The definitions as per the search criteria will be filtered and shown on the summary page. You can specify one or multiple search criteria in this section.

Figure 24-2 Search Criteria



Copyright © 2011, 2026, Oracle and/or its affiliates.

The Data Creation Process summary Window displays the following columns:

- **Name**: Displays the Data Creation Process Rule's name.
- **Folder**: Displays the Folder name where the Data Creation Process Rule is saved.
- **Access Type**: Displays the access type of rule. It can be Read-Only or Read/Write.
- **Creation Date**: Displays the Date and Time when Data Creation Process was created.
- **Created By**: Displays the Name of the user who created the Data Creation Process.
- **Last Run By**: Displays the Name of the user who last runs the Data Creation Process.
- **Last Run Date**: Displays the Date and Time when Data Creation Process was Run last.
- **Status**: Displays the status of the Rule.
- **Action**: Displays the list of actions that can be performed on the Data Creation Process Rule. For more information, see the Data Creation Process rule – Icons and Descriptions.

The Action column on the Data Creation Process summary page offers several actions that allow you to perform different functions. The following actions are available for the Data Creation Process Rule.

- **Add**: Click the Add icon from Top right corner to create a new Data Creation Process Rule.
- **Delete**: Click the Delete icon from Top right corner to create a new Data Creation Process Rule.
- **Refresh**: Click the Refresh icon from Top right corner to refresh the Data Creation Process Rule.
- **View/Edit**: Click in the Action column and select View/Edit to view or edit the contents of a Data Creation Process rule in Read/Write Mode.
- **Save As**: Click in the Action column and select Save As to create a copy of an existing Data Creation Process Rule.
- **Delete**: Click in the Action column and select Delete to delete an existing Data Creation Process Rule.
- **Execute**: Click in the Action column and select Execute to execute an existing Data Creation Process rule. For more information, see Executing Data Creation Process section.
- Creating Data Creation Process
  To create a new Data Creation process, perform the following steps:
- Executing Data Creation Process

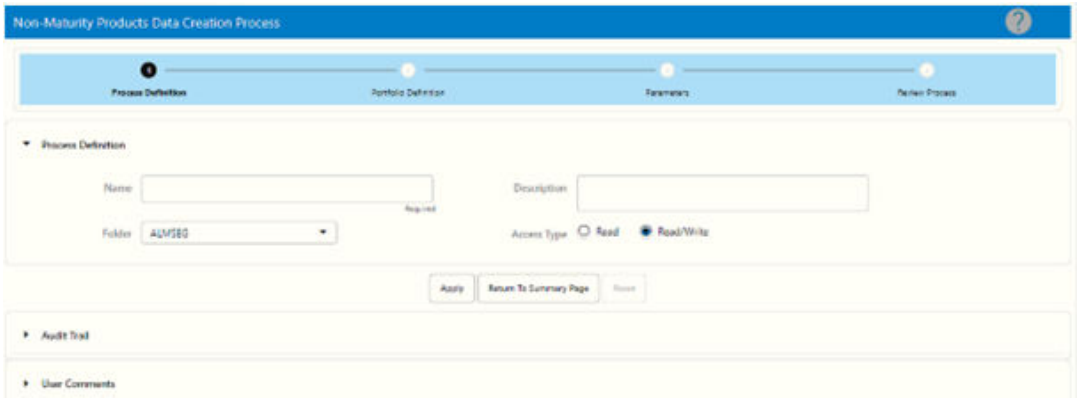
24.1 Creating Data Creation Process

To create a new Data Creation process, perform the following steps:

1. Navigate to the **Data Creation Process** summary page.
2. Click **Add**. The Create Data Creation Process page is displayed.

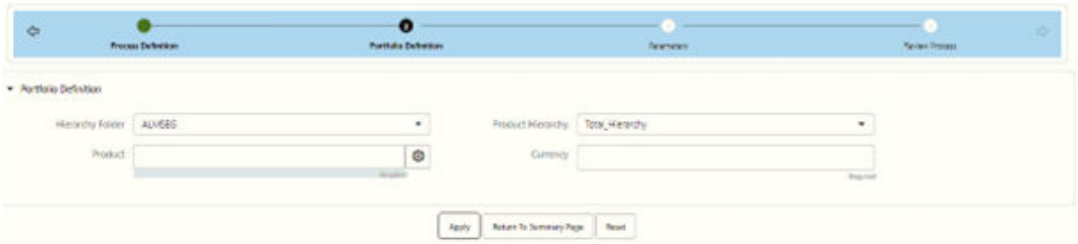
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 24-3 Creating Data Creation Process



3. Enter the following details.
   - **Name**: Enter the name of the Data Creation Process Rule. The Data Creation Process Name should be unique. Any special characters are not applicable.
   - **Description**: Enter the description of the Data Creation Process Rule.
   - **Folder**: Select the Folder where the Data Creation Process needs to be saved.
   - **Access Type**: Select the Access Type as Read-Only or Read/Write.
4. Click **Apply** to navigate to the Product Selection section.

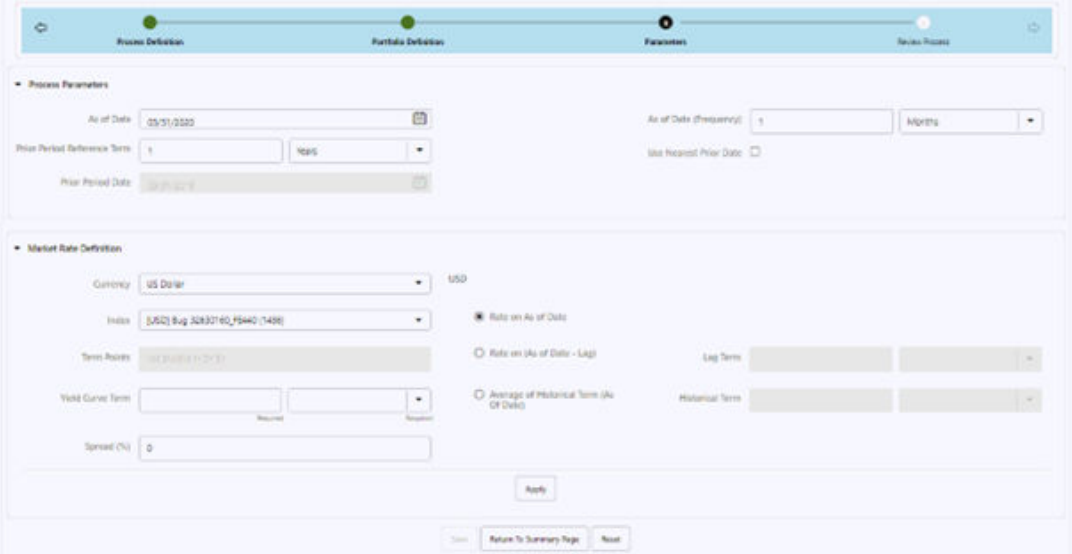
Figure 24-4 Product Definition section



5. Enter the following Portfolio Selection details:
   - **Hierarchy (Folder)**: Select the Folder from which you want to pick the Hierarchy.
   - **Product Hierarchy**: You can specify some processing parameters at product-currency combination. Hierarchies in the selected Folder will be listed and you can select one from the available list of Hierarchies.
   - **Product**: Select the product. This field shows the Product list based on any node/leaf or any combination.
   - **Currency**: Select the Currency. The Currency drop-down displays the list of active currencies. You can select multiple currencies as well.
6. Click **Apply** to navigate to the Parameters section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 24-5 Parameters section



7. Enter the Parameters as shown in the following table:

Table 24-1 List of Parameters details used for Creating Data Creation Process

Parameter	Description
As of Date	As of Date and prior period reference term allows you to define the period. By default, As of Date is picked from Application Preferences. Here, you can change this date if required.
Prior Period Reference Term	As per the prior period reference term (default value 1 Year), the prior period date is auto calculated.
Prior Period Date	Displays the Prior Period Date. You cannot edit this field as it is calculated as per the Prior Period Reference Term.
As of Date (Frequency)	The Default value of this field is 1 month. As of Date (Frequency) is provided to select any value in days, months, or years as 1 Day, 3 Months, 1 Year, and so on. So, starting prior period date, data snaps will be taken at defined snap frequency till As of Date. For each date, EOD balance (CUR_PAR_BAL) for the account on Data Snap Date, Account Origination Date (which will be required to calculate account age/vintage at each Data Snap Date), and Deposit Rate (CUR_NET_RATE) will be captured, along with market-rate as per the index selected.
Use Nearest Prior Date	If Use Nearest Prior Date checkbox is selected, the system picks the nearest date (current period as of date) as per the last successful Run (default checked). If it is selected, then you cannot edit the prior period reference term field and the Prior Period Date is picked from the last successful Run for the same Data Creation Process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 24-1 (Cont.) List of Parameters details used for Creating Data Creation Process

Parameter	Description
Market Rate Definition	The Market Rate Definition parameters allow you to define the Market Rate Parameters such as Index, Currency, Term Points, Yield Curve Term, Spread, and IRC Effective Date. • **Currency**: This will list down all the currencies for which the portfolio is defined in portfolio definition UI. • **Index**: As per the selected currency from the drop-down; all the interest rate curves defined will be given for selection. Users can choose one from which market rate will be picked. • **Term Points**: Shows the term points. • **Yield Curve Term**: As per the selected interest rate curve, the user can choose one term point, the rate corresponding to the selected term point will be picked. For intermediate-term points interpolation is done. • **Spread**: Default value is 0. This field accepts all positive, negative integers/decimals in the range of -100% to +100%), you can enter the spread in a percentage format. Enter the spread in percentage on top of the rate picked from the selected IRC and term point. • **IRC Effective Date**: Rates will be picked as per the IRC effective date selected on one of the following options: – **Rate on As of Date**: Effective Date is same as As of Date. – **Rate on (As of Date- Lag)**: Effective date is As of Date minus some lag. If you select Rate on (As of Date- Lag), then the Lag Term field will be enabled to enter term and tenor. – **Average of Historical Term (As Of Date)**: Effective date is not just one date, but a period starting As of Date; over which arithmetic average will be taken to smooth out any interest rate fluctuations If you select **Average of Historical Term (As Of Date)**, then the **Historical Term** field will be enabled to enter term and tenor.

8. Click **Apply** and navigate to the Review Process section.
   Here, you can review all the entered details in the preceding stages and edit them if required.
9. Verify the changes and click **Apply**.

24.2 Executing Data Creation Process

The **Execute** option will run the Data Creation Request and create data as for the selected portfolio.

Prerequisites

Predefined Data Creation Process

Procedure

Following two approaches are available to execute the process:

- UI Based Procedure
- Batch Based Procedure
- UI Based Procedure
- Batch Based Procedure

Copyright © 2011, 2026, Oracle and/or its affiliates.

24.2.1 UI Based Procedure

To execute the Data Creation Process, follow these steps:

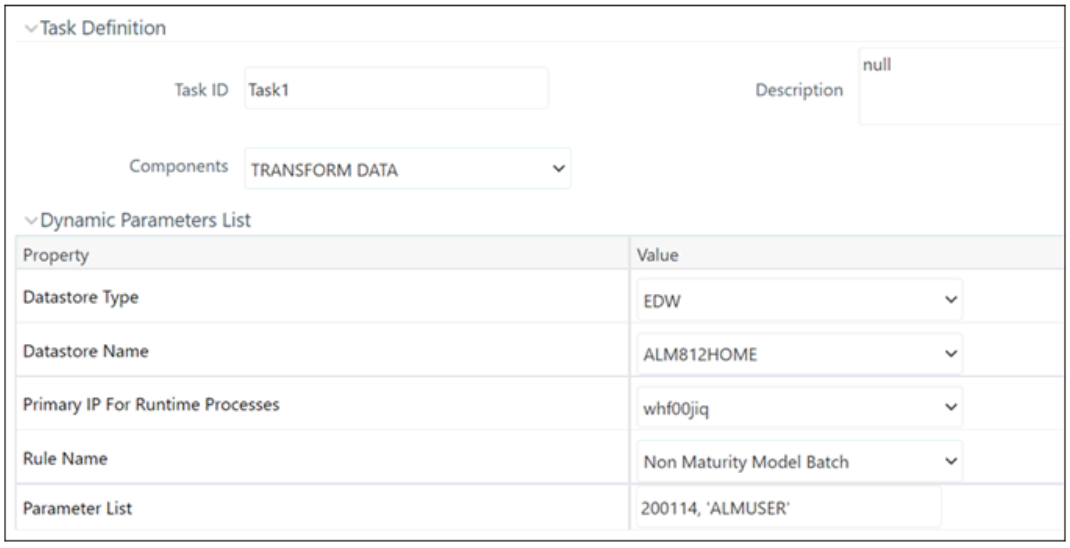
1. Navigate to the **Data Creation Process** Summary Page.
2. Search for a Data Creation process Rule.
3. Click the **Action** column and select **Execute** to execute an existing Data Creation Process Rule.
4. Click **Run**.
5. The Data Creation Process Run Confirmation Page is displayed. The status of the process is displayed in the Status column.
   After completion of the process, you can navigate to the **Execution Details** Page by selecting the view log option. From the Execution Details, you can access a report that provides details of any processing errors that were encountered while running the data creation process.

24.2.2 Batch Based Procedure

Batch Based Procedure To execute the batch with Run Rule Framework, follow these steps:

1. From the LHS menu, select **Common Object Maintenance**, select **Operations**, and then select **Batch Maintenance**.
2. Create a new batch.
3. Select the Batch Name to add the task.
4. Click the **Add** button under the Task Details section.

Figure 24-6 Task Details section



5. Define the Task ID and Description.
6. Select Components as **TRANSFORM DATA**.
7. Input the following required parameters:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Datastore Type:** Select appropriate datastore from list
- **Datastore Name :** Select appropriate name from the list
- **Primary IP for Runtime Processes:** Select the IP address from the list
- **Rule Name:** Non Maturity Model Batch
- **Parameter List:** System ID, User Name

Enter the parameters in the following format: 200114, ‘ALMUSER’

Here, 200114 is System ID and ALMUSER is User Name

8. Save the rule and **Execute**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

25

Non Maturity Products Model Analysis

Non-Maturity Products Modeling helps you to understand the relation between market and bank rate, how much is core out of total available balance for the portfolio, and how long core will remain with the bank based on Decay Rate Profile

The Non-Maturity Products Model Analysis Summary Page allows you to do the following:

- Create Products Model Analysis Rule
- Searching for Products Model Analysis Rule. For more information, see the Searching for Rules section
- Viewing and Updating Products Model Analysis Rule. For more information, see the Viewing and Editing Rules section.
- Copying Products Model Analysis Rule. For more information, see the Copying Rules section.
- Deleting Products Model Analysis Rule. For more information, see the Deleting Rules section.

The definitions based on the search criteria are listed under the List of Non-Maturity Products Model Analysis. This is the search section that contains multiple parameters. You can specify one or multiple search criteria in this section. When you click Search, depending upon the search criteria, it filters and displays the list of Non-Maturity Products Models meeting the search criteria. The Non-Maturity Products Model Analysis Summary Window displays the following columns.

Column	Description
Name	Displays the Non-Maturity Products Model Analysis Rule's short name.
Currency	Displays the Currency for which Non-Maturity Products Model Analysis Rule is defined. One rule can be created only on one particular currency.
Folder	Displays the Folder name where the Non-Maturity Products Model Analysis Rule is saved.
Created By	Displays the Name of the user who created the Non-Maturity Products Model Analysis Rule.
Creation Date	Displays the Date and Time when Non-Maturity Products Model Analysis Rule was created.
Last Modified By	Displays the Name of the user who modified last the Non-Maturity Products Model Analysis Rule.
Last Modified Date	Displays the Date and Time when Non-Maturity Products Model Analysis Rule was modified last.
Status	Displays the status of the rule. Complete, In Progress, or Failed.
Action	Displays the list of actions that can be performed on the Non-Maturity Products Model Analysis rule. For more information, see Non-Maturity Products Model Analysis Rule- – Icons and Descriptions.

Copyright © 2011, 2026, Oracle and/or its affiliates.

The Action column on the Non-Maturity Products Model Analysis Summary Page offers several actions that allow you to perform different functions. The following actions are available for the Non-Maturity Products Model Analysis Rule.

Table 25-1 Table: Non-Maturity Products Model Analysis Rule – Icons and Descriptions

Fields	Description
Add	Click **Add** icon to create a new Non-Maturity Products Model Analysis Rule.
Delete	Click Delete icon from Top right corner to delete the Data Creation Process Rule.
Refresh	Click Refresh icon from Top right corner to refresh the Data Creation Process Rule.
View	Click the **Action** column and select **View** to view the contents of a Non-Maturity Products Model Analysis Rule in Read mode.
Edit	Click the **Action** column and select **Edit** to edit the contents of a Non-Maturity Products Model Analysis Rule in Read/Write mode.
Save As	Click the **Action** column and select **Save As** to create a copy of an existing Non-Maturity Products Model Analysis Rule.
Delete	Click the **Action** column and select **Delete** to delete an existing Non-Maturity Products Model Analysis Rule.
Dependency Check	Click the **Action** column and select Check Dependency to check the dependency of an existing Non-Maturity Products Model Analysis Rule.

- Searching for a Non-Maturity Products Model Analysis Rule
- Creating Non-Maturity Products Model
- Editing Non-Maturity Products Model Analysis Rule
- Deleting Non-Maturity Products Model Analysis Rule
- Copying Non-Maturity Products Model Analysis Rule
- Dependency Checking

25.1 Searching for a Non-Maturity Products Model Analysis Rule

Search for a Model to perform any of the following tasks:

- Update, Copy or Delete existing rules.
- Define methodologies for products or define other model parameters.

To search for a model, follow these steps:

1. Navigate to the Non-Maturity Products Model Analysis Window.
2. Enter the search criteria, one or all of following details:
   - Name
   - Folder
   - Dimension

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Currency
- Status

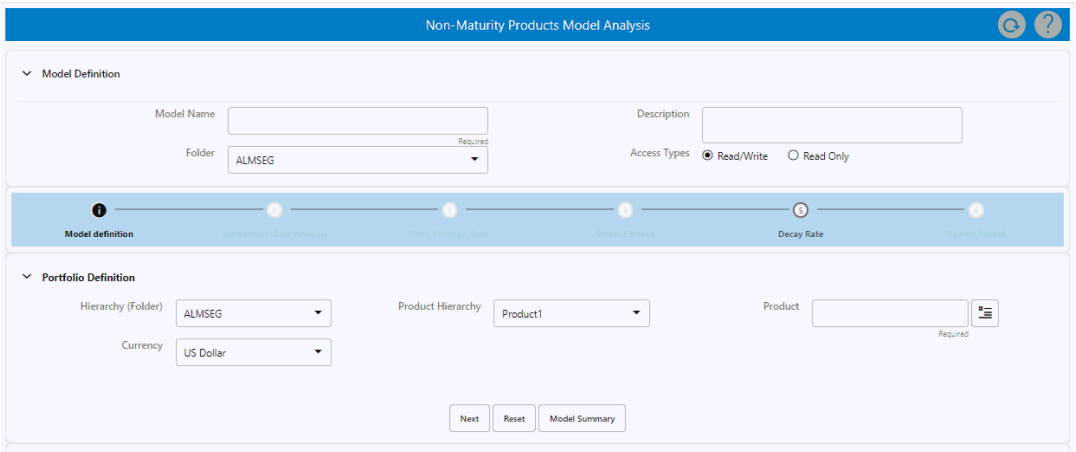
3. Click Search. Only rules that match the search criteria are displayed.

25.2 Creating Non-Maturity Products Model

To create a new Non-Maturity Products Model Analysis Rule, perform the following steps:

1. Navigate to the Non-Maturity Products Model Analysis Summary Page.
2. Click **Add** . The Create Non-Maturity Products Model Analysis Page will be displayed.

Figure 25-1 Creating Non-Maturity Products Model Analysis Rule



3. Enter the details shown in the following table.

Table 25-2 List of process details used for Creating Non-Maturity Products Model Anal

Fields	Description
Model Name	Enter the name of the Non-Maturity Products Model Analysis Rule. This should be unique name. This field accepts Alpha Numeric value. Special characters are not allowed.
Description	Enter the description of the Non-Maturity Products Model Analysis Rule. The maximum limit of this field is 300 characters. You can enter special characters in this field.
Folder	Select the folder where you want to save the Non-Maturity Products Model Analysis Rule.
Access Type	Select Access Type as Read-only or Read/Write.
Hierarchy (Folder)	Select the folder where product is stored on which model need to be build.
Product Hierarchy	Select the Product Hierarchy from available list of hierarchies in selected folder.

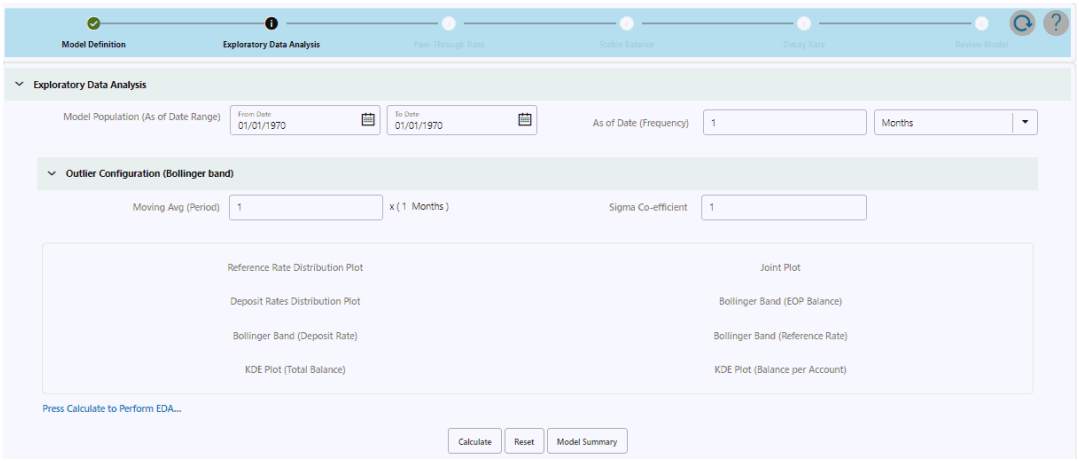
Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 25-2 (Cont.) List of process details used for Creating Non-Maturity Products Model Anal

Fields	Description
Product	Based on the selected product hierarchy in model definition UI, Hierarchy Browser will expand the product hierarchy and you can select any parent/leaf node by clicking on the view button next to the product in the model definition screen.
Currency	Select the currency for the Non-Maturity Products Model Analysis Rule. Only one currency can be selected for a particular model.

4. Click **Apply** to navigate to the **Exploratory Data Analysis** section.

Figure 25-2 Creating Non-Maturity Products Model Analysis Rule - Exploratory Data Analysis section



The **Exploratory Data Analysis** Window will help you to perform EDA calculations and get a glimpse of data being used for modelling.

5. Enter the following details in the **Exploratory Data Analysis** window.

Table 25-3 Fields on Exploratory Data Analysis

Fields	Description
Model Population (As of Date Range)	You can select a date range which will define the historical period within which data needs to be picked for Model creation By default, the Date range would be populated as below: End Date: Max (As of Date) available in Model input Table. Start Date: Max [min (As of Date), End Date – 10 Years]. Default values are populated from FSI_AUDIT_NM_LOAD_BATCH table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

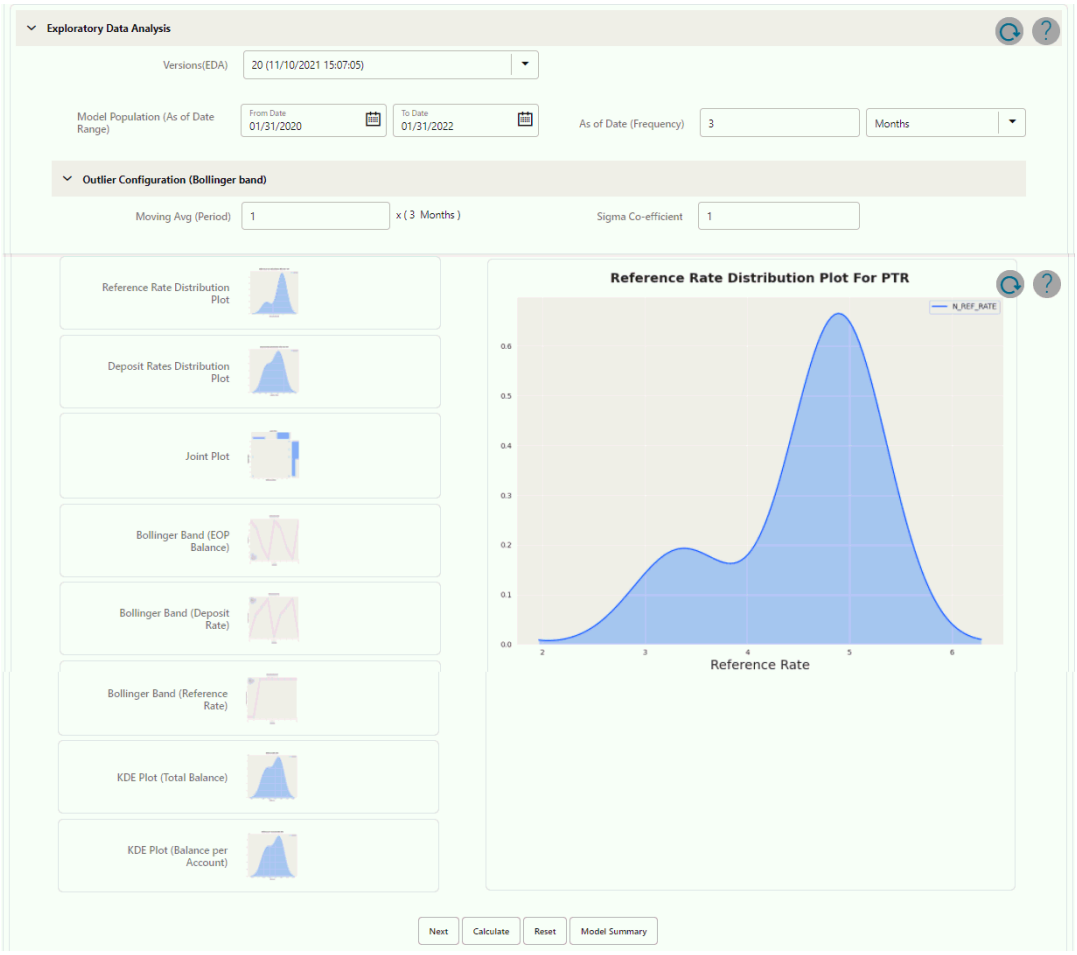
Table 25-3 (Cont.) Fields on Exploratory Data Analysis

Fields	Description
As of Date (Frequency)	The default value of this field is one month. As of Date (Frequency) defines on what regular intervals, data needs to be picked from the input table. For example, you can have historical period as last three years, in those three years at what regular interval data needs to be picked up from history table that would be defined by As of Date (Frequency).
Moving average period for Bollinger Band	The default value of this field is one month. The Moving Average period defines the historical term over which the moving average needs to be calculated for Bollinger band formation.
Sigma Co-efficient (Bollinger Band)	The default value of this field is 1. Sigma Co-efficient defines the outliers, any number which is (sigma coefficient * standard deviation) away from the mean, will not be considered for calculations. This field accepts Integer and Decimal values in the range 0 – 10.

6. Click **Calculate** to perform Exploratory Data Analysis and get EDA plots. You can refresh the window using the Refresh button to check if the EDA is completed. When you open EDA window first time for a particular model, Version(EDA) field would not appear as there is no previous version to display. Subsequently, when first version is created and you go for second version, version (EDA) tab will start appearing with previous EDA versions. When EDA version is submitted, a message will appear with version number. You get a pop-up in case last run version was failed. If run was successful, latest version would appear on the UI, when you log back in. **Reset** button resets the entered values to their default values.

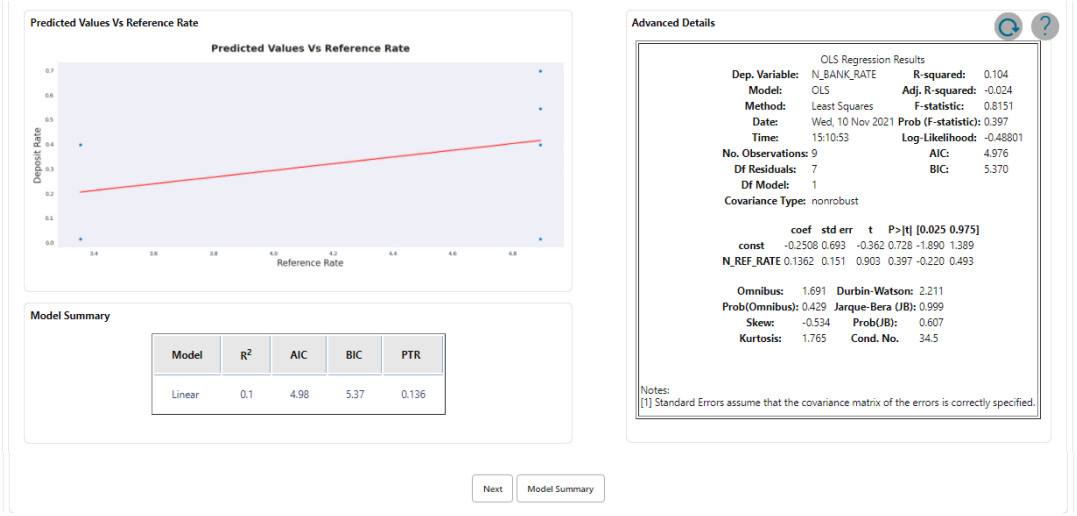
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 25-3 Exploratory Data Analysis section



7. Click Next to navigate to the **Pass-Through Rate Calculations** window.

Figure 25-4 Pass-Through Rate Calculations

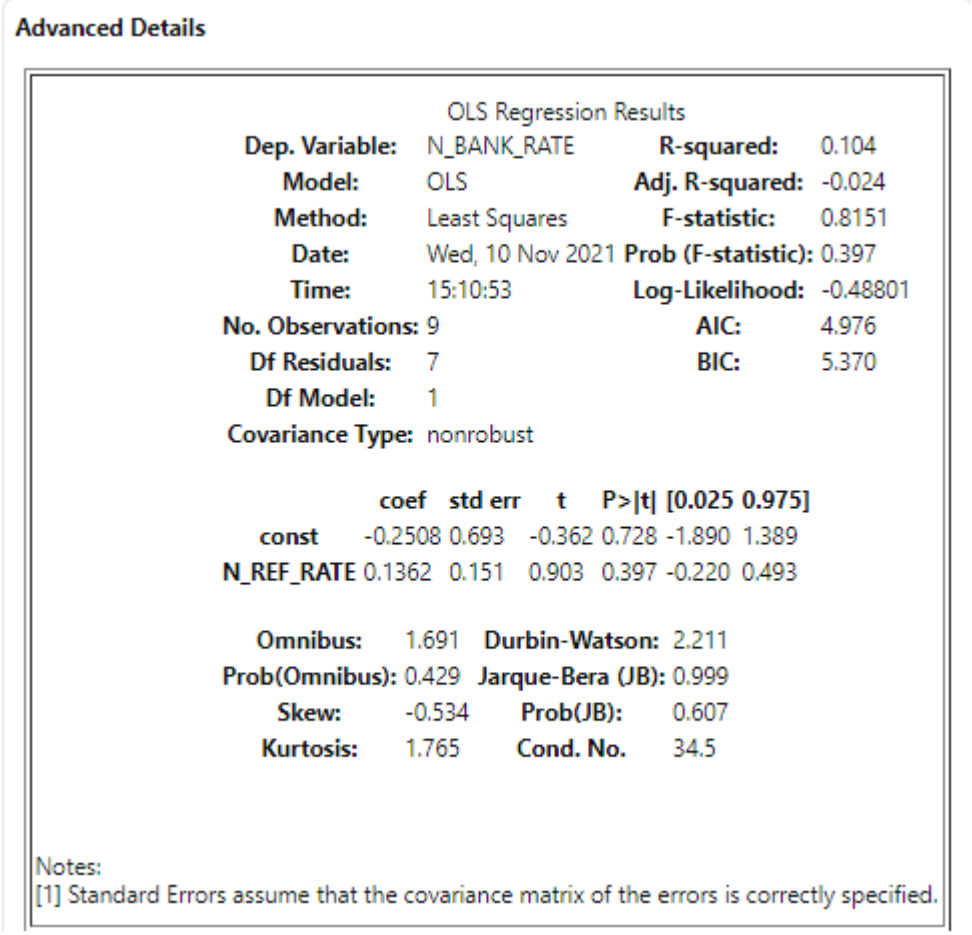


8. Click **Calculate** to view all the evaluation parameters and quality plots. Pass-Through Rate calculations are submitted, a message appears with version number. If you come back after some time, you get a pop-up in case last run version was failed. This window shows

Copyright © 2011, 2026, Oracle and/or its affiliates.

all the model quality parameters and graphs. At this stage, Pass through rate or beta between market rate and deposit rate is calculated. You can zoom the graph to enhance visibility. Following Advanced Details Window will help you to evaluate the model fit.

Figure 25-5 Advanced Details


Advanced Details

OLS Regression Results

Dep. Variable:	N_BANK_RATE	**R-squared:**	0.104
Model:	OLS	**Adj. R-squared:**	-0.024
Method:	Least Squares	**F-statistic:**	0.8151
Date:	Wed, 10 Nov 2021	**Prob (F-statistic):**	0.397
Time:	15:10:53	**Log-Likelihood:**	-0.48801
No. Observations:	9	**AIC:**	4.976
Df Residuals:	7	**BIC:**	5.370
Df Model:	1		
Covariance Type:	nonrobust		

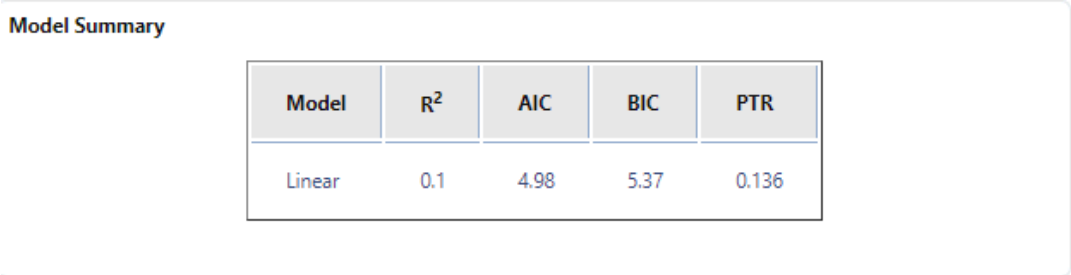
	coef	std err	t	P>|t|	[0.025	0.975]
const	-0.2508	0.693	-0.362	0.728	-1.890	1.389
N_REF_RATE	0.1362	0.151	0.903	0.397	-0.220	0.493

Omnibus:	1.691	**Durbin-Watson:**	2.211
Prob(Omnibus):	0.429	**Jarque-Bera (JB):**	0.999
Skew:	-0.534	**Prob(JB):**	0.607
Kurtosis:	1.765	**Cond. No.**	34.5

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

The Model parameters like R^2, AIC, BIC and regression co-efficient will be shown under model summary.(how much change in market rate is passed on to the deposit rate).

Figure 25-6 Model Summary


Model Summary

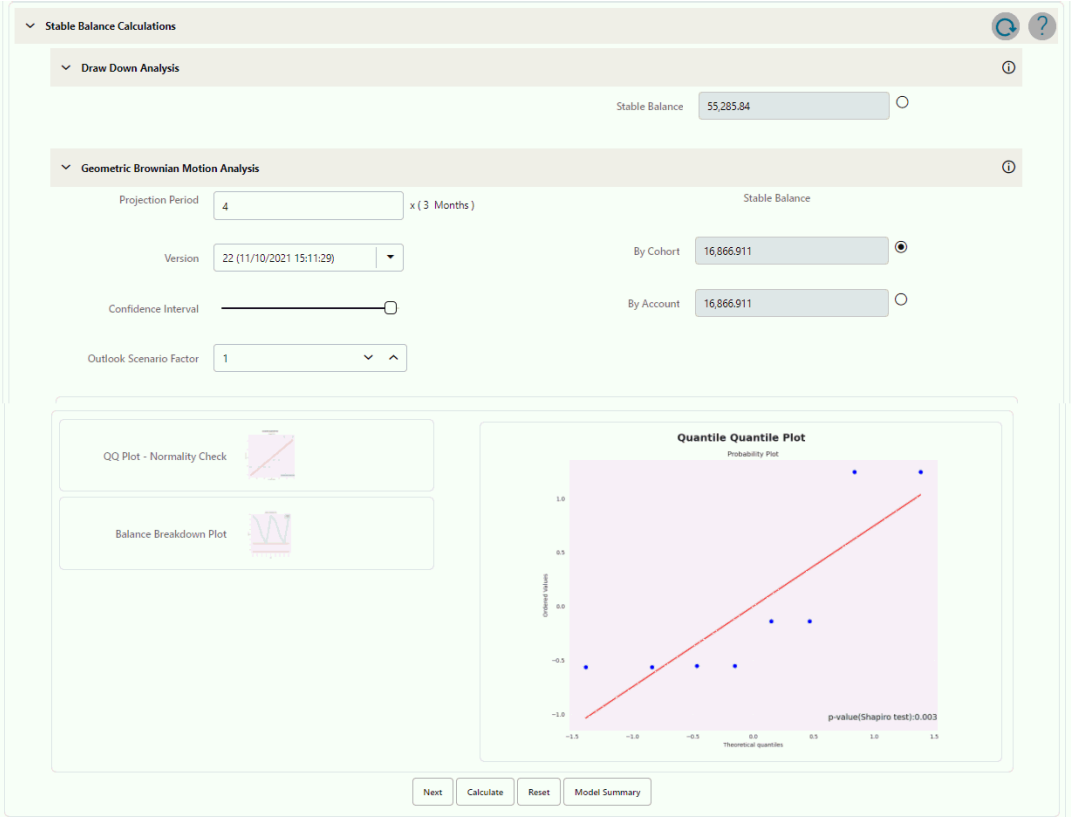
Model	R^2	AIC	BIC	PTR
Linear	0.1	4.98	5.37	0.136

Copyright © 2011, 2026, Oracle and/or its affiliates.

At any stage if user would like to come out of the model analysis module, one can Click on Model Summary button and go back to Non-Maturity products model summary screen and come out of modelling module.

9. Click **Next** to navigate to the Stable Balance Calculation section. This window shows the Projection period. You can set it as required. Projection period can be only be positive integers which will act as multiplier to As of Date (Freq). E.g. if as of date (Freq) is quarter and projection period is 8 * As of Date (Freq), it implies projection period is 2 years.

Figure 25-7 Stable Balance Calculation



10. To calculate Stable Balance, you can use following methods:

- Draw Down Analysis
- Geometric Brownian Motion

Draw Down Analysis: The Draw Down Analysis calculates each period's Run-offs in the historical period and subtracts the maximum Run-off from the most recent total balance to arrive at stable balance.

Geometric Brownian Motion: This approach will first calculate the past Run-offs volatility and assuming normal distribution of Run-offs, accept confidence interval as user input, projection period for time adjustment, and outlook scenario factor (volatility multiplier) as input from the user. After clicking calculate, the stable balance would be shown at cohort/ account level using both cohort level and account level volatility and saved against version.

The confidence interval slider field has a range of 0 – 100% and by default value as 95% and Outlook scenario factor (Volatility Multiplier) is set at 1 (Default Value).

The range of outlook scenario factor is 0.01 to 1.99.

You can change any of the inputs and generate multiple versions.

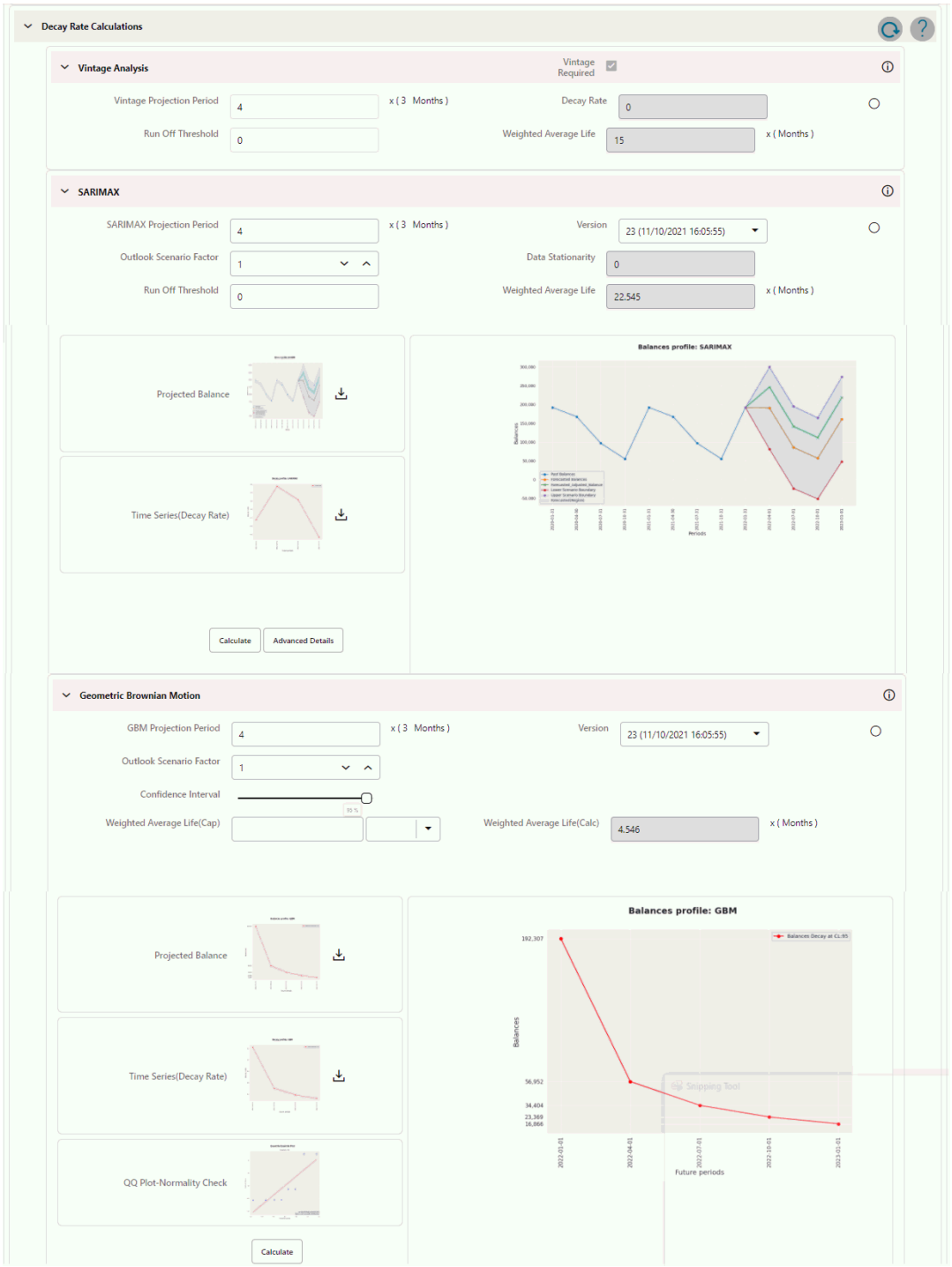
Copyright © 2011, 2026, Oracle and/or its affiliates.

A radio button is available for each method. Stable balance can be calculated with multiple methods, so before proceeding to next step, you must select one radio button to confirm one stable balance on which rest of the subsequent processing will take place. When this window is displayed first time, after pressing the Calculate, stable balance is calculated for both methods, Drawdown analysis and Geometric Brownian Motion. In subsequent runs, if any of projection period, confidence interval or Scenario factor is changed, then new version will be generated only for GBM. When you click **Calculate** button, then a pop up is displayed: " GBM version xyz is submitted'. If you come back and version is failed, a pop up is displayed: "'GBM version xx is failed' and UI will revert back to last successful version".

11. Click **Next** to navigate to the **Decay Rate** Calculations Section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 25-8 Decay Rate Calculations



For Decay rate calculations, you can choose any one of following three approaches:

- Vintage Analysis
- SARIMAX
- Geometric Brownian Motion

Copyright © 2011, 2026, Oracle and/or its affiliates.

Vintage Analysis: This method calculates one decay rate value as per the various vintages and regular intervals over which data is being picked for the defined portfolio. The same decay rate is applicable to the whole projection period. Select Vintage Analysis checkbox if you want to use this in calculation or exclude it from calculations. Vintage analysis needs account-level data for complete historical period, which you may not have initially, so you can avoid this method. For other methods, portfolio/cohort level data is required which is comparatively easier to get. By default, this checkbox is not selected to indicate this model is not included in processing. . You can set the projection period as per your requirement.

SARIMAX: This method creates a future balance profile based on historical balances, along with Run-off volatility multiplied by the Outlook scenario factor (default 1). Based on balance Run-offs, a decay profile is created.

Geometric Brownian Motion: Based on balance Run-off volatility and accepted user inputs like confidence interval and outlook scenario factor, Decay Rates are calculated for each subsequent period in projection period and decay profile is created.

Based on the Decay Rate Profile, Run-offs would be calculated and finally, a balance profile will be created which will in turn help to calculate WAL.

Based on the Decay Rates profile, the model will do Weighted Average Life (WAL) calculations as well, if WAL is more than defined regulatory WAL (Cap), then the system will let the user define a confidence interval range

Figure 25-9 Confidence Interval Range



and steps (Default 5%) in which confidence interval will be increased for each calculation. If the range is 80-95%, then the system generates a Decay Rate Profile at 80%, 85%, 90% and 95% interval; using an iterative process, try to calculate a Decay Rate Profile which will give WAL within regulatory WAL (Cap) defined by the user. If in a given confidence interval range, WAL is still more than regulatory WAL (Cap), a simple linear Decay Rate Profile would be created.

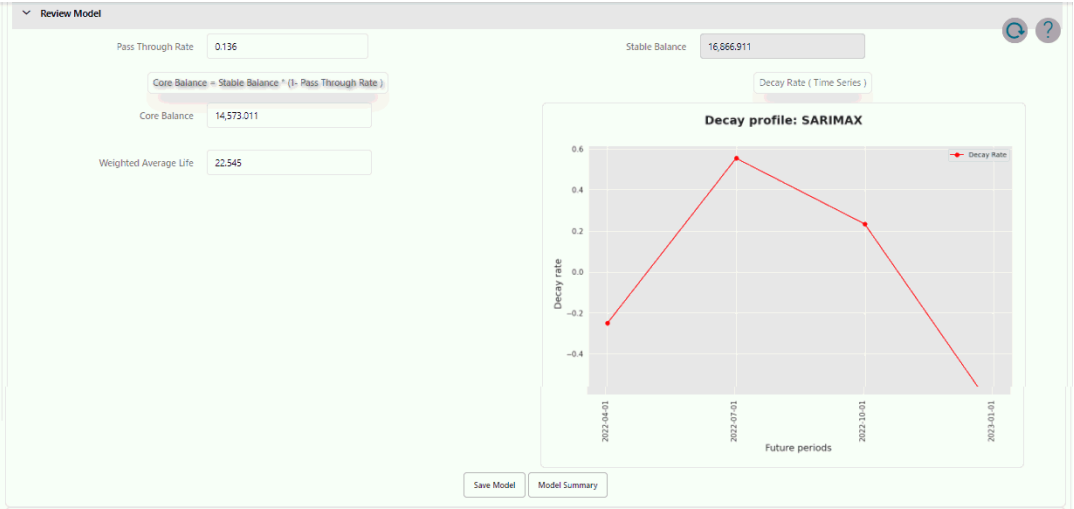
Run-off threshold is a user input, it will define the minimum balance beyond which WAL calculations would not be performed, considering balance become zero at that point.

When this window is launched first time and you press Calculate, Decay Rate is calculated for all three methods, Vintage Analysis, SARIMAX and Geometric Brownian Motion. After first run (when Calculate button does not appear), separate calculate buttons (as shown with SARIMAX and Geometric Brownian Motion) are available for each of Geometric Brownian motion and SRIMAX.

12. Click **Next** to navigate to the **Review Model** Section. This section shows the Model details before confirming the model. Here, you cannot edit any details. If you want to update any details, click Model Summary to go back. Until model is saved, it is always be in *Draft* status. It converts to *Complete* only when you save the model. For more information on editing, see the Editing Non-Maturity Products Model Analysis Rule section.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 25-10 Review Model section



25.3 Editing Non-Maturity Products Model Analysis Rule

To view and edit a Non-Maturity Products Model Analysis Rule, follow these steps:

1. Navigate to the Rule Summary Page of the rule you want to update.
2. Search for a Rule. For further information, see the Searching for Rules section.
3. Click in the Action column and select View/Edit to open the rule you want to update.
4. Update the rule details.
   - If model is Confirmed, then only Stable Balance and Decay Rate profile can be updated.
   - If the model is in Draft, then you can edit all the details except Name, Access Type and Folder fields.
5. Click Apply or Save, depending on the Model Status.

25.4 Deleting Non-Maturity Products Model Analysis Rule

You can delete rules that are no longer required.

> **Note**
>
> A rule cannot be retrieved after deletion.

Restrictions on deleting rules are:

- You cannot delete rules if you have only Read privileges. Only users with read/write privilege and rule owners can delete model rules.
- You cannot delete a rule that has a dependency that is model is being used in any of the Behavior Patterns.

To delete a Non-Maturity Products Model Analysis rule, follow these steps:

1. Navigate to the model Rule Summary Page and select the Rule you want to delete.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Click in the Action column and select Delete.

25.5 Copying Non-Maturity Products Model Analysis Rule

You can copy rules to avoid having to enter data multiple times. This saves time and effort, and also reduces operational errors. To copy a Non-Maturity Products Model Analysis Rule, follow these steps:

1. Navigate to the model Rule Summary Page.
2. Search for a Rule. For more information, see the Searching for Rules section.
3. Click in the Action column and select Save As to duplicate the rule.
4. Select a folder where you want to save the rule copy.
5. Enter a unique name for the new rule. (Optional) Enter a brief description of the rule
6. Select the access type.
7. Click the Save button.

25.6 Dependency Checking

You can check dependencies for rules to know where a particular model rule is being used. This also prevents the accidental deletion of rules having dependencies.

To check the dependency of the Non-Maturity Products Model Analysis Rule, follow these steps:

1. Navigate to the Model Rule Summary Page
2. Search for a rule. For further information, see the Searching for Rules section.
3. Click in the Action column and select Dependency Check to the rule that you want to check for.

Copyright © 2011, 2026, Oracle and/or its affiliates.

26
Adjustment Rules

Adjustment Rules allow you to specify methodologies to calculate Transfer Pricing add-on rates and breakage charges for the relevant products in your portfolio. The Standard Transfer Pricing Process references the methodologies contained in the Adjustments rule.

The procedure for working with and managing Adjustment rules is similar to that of other Oracle Funds Transfer Pricing business rules. It includes the following steps:

- Searching for Adjustment Rules
- Creating Adjustment Rules
- Viewing and Editing Adjustment Rules
- Copying Adjustment Rules
- Deleting Adjustment Rules

As part of creating and editing Adjustment Rules, you define Adjustment Methodologies for applicable products.

- Defining Adjustment Methods
- Availability of Adjustment Methods
- Adjustment Method Parameters
- Procedure to Define the Formula Based Rate Adjustment Method

Oracle Funds Transfer Pricing provides you with the option to copy, in total or selectively, the product assumptions contained within the Adjustments, Transfer Pricing, and Prepayment Rules from one currency to another currency or a set of currencies or from one product to another product or a set of products.

- Create Adjustment Rules
  You create an Adjustment Rule to define Adjustment Methodologies for your products.
- Defining Adjustment Methods
  The definition of Adjustment Methods is part of the Create or Edit Adjustments Rule process.

26.1 Create Adjustment Rules

You create an Adjustment Rule to define Adjustment Methodologies for your products.

To create the Adjustment Rule:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Assumption Specification**, and then select **Adjustment Rules**.

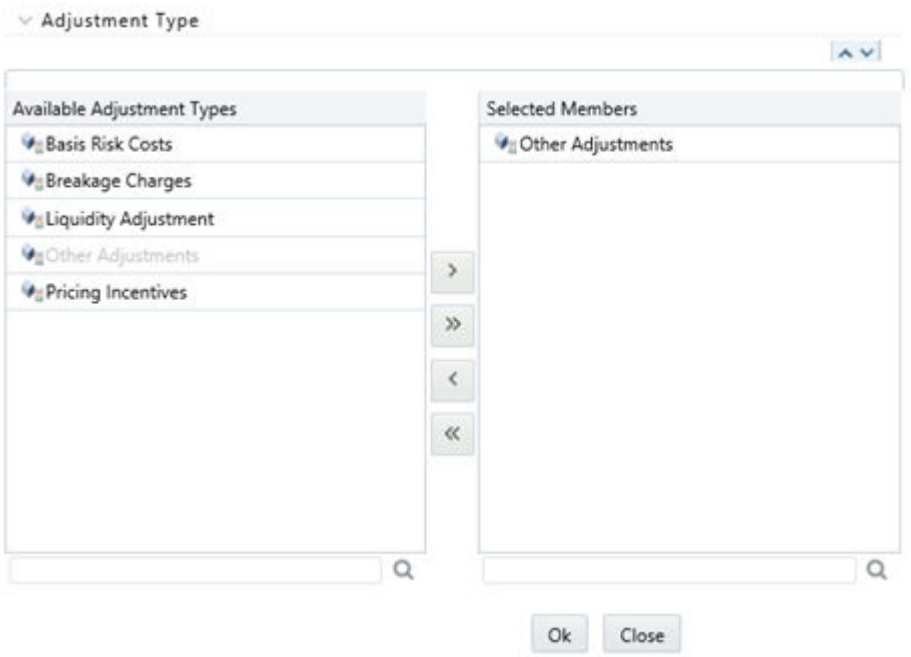
Figure 26-1 Adjustment Rule - Definition Mode



Copyright © 2011, 2026, Oracle and/or its affiliates.

2. Select the Adjustment Type from the Adjustment Type Selection section.

Figure 26-2 Adjustment Type Selection page



3. Complete standard steps for this procedure.

26.2 Defining Adjustment Methods

The definition of Adjustment Methods is part of the Create or Edit Adjustments Rule process.

When you click Save in the Create Adjustments Rule Process, the rule is saved and the Adjustments Rule Summary Page is displayed. However, adjustment assumptions have not yet been defined for any of your products at this point. You would start defining your adjustment assumptions for product-currency combinations before clicking Save. To define an Adjustment Method, follow these steps:

1. Perform the basic steps for creating or editing an Adjustment Rule.
2. Populate the following details:

Table 26-1 Fields and Descriptions from the Adjustment Specification screen

Term	Definition
Reference Term	The associated term is used for the Add-on Rate assignment. You can select one of the following types of reference terms: Original Term (the contractual term to the maturity of the account), Repricing Frequency (the frequency at which the account reprices), Remaining Term (the number of months until the account matures).
Interest Rate Code	Used for the Rate Lookup for the Formula Based Rate, and in the Breakage Charge - Economic Loss Method when discounting cash flows.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 26-1 (Cont.) Fields and Descriptions from the Adjustment Specification screen

Term	Definition
Assignment Date	Allows you to choose the date for which the yield curve values are to be sourced. Choices available are the As-of-Date, Last Repricing Date, TP Effective Date, Origination Date, Commitment Start Date, or Adjustment Effective Date.
Lookup Method	The method used to derive an Add-on Rate for different reference term values. • Specify **Range** as the Lookup Method if you want the application to apply the rates defined in the Adjustments rule to a range of reference term values, using the terms defined in the rule to specify the lower end of the range. Note that for values less than the lowest term point, the application uses the value associated with the lowest point. • Specify **Interpolation** as the Lookup Method if you want the application to Interpolate Add-on Rate values for applicable reference terms falling between node points specified in the Adjustments Rule, using straight-line interpolation between the term points. Deals that are outside of range or ranges will not be populated with any values.
Term	In conjunction with the Multiplier, this field allows you to specify the value for the Reference Term, for a given Lookup Tier.
Multiplier	The unit of time applied to the Term. The choices are Days, Months, Years.
Rate	The Add-on Rate to be applied to instruments where the Reference Term is the product of the Term and Multiplier defined for the row. The rate should be in percentage form, for example, 1.25 percent should be input as 1.25.
Amount	The Add-on Amount to be applied to instruments where Reference Term is the product of Term and Multiplier defined for the row.
Formula	The mathematical formula used in the Formula Based Rate Adjustment Method to determine the Add-on Rate: (Term Point Rate * Coefficient) + Rate Spread.
Rate Floor and Rate Cap	The minimum and maximum rate. If the calculated value is less than the floor or more than the cap, then these rates will be applied. These boundaries are applicable only to Formula Based Method and Use the TP method from TP Rule Adjustment Method types. These are optional inputs. Ensure that the Rate Floor value is always less than or equal to the Rate Cap Value.
Term Point	In conjunction with the Multiplier (day, month, or year), it is used in the Formula Based Rate Method when looking up the rate for the designated Interest Rate Code.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 26-1 (Cont.) Fields and Descriptions from the Adjustment Specification screen

Term	Definition
Coefficient	Coefficient by which the Term Point Rate should be multiplied.
Rate Spread	The spread added to the Interest Rate read from the selected Interest Rate Code. Rate Spread is used in the Formula Based Rate and Breakage Charge - Economic Loss Adjustment Methods. For the Formula Based Rate Method, the spread is added to the result of the Term Point Rate * Coefficient. Enter the Rate Spread in percentage form, for example, 1.25 percent should be input as 1.25.
Minimum Charge	Used in the Fixed Percentage and Economic Loss Adjustment Methods for Breakage Charges. If the calculated Break Funding Amount is less than the Minimum Charge, then the Minimum Charge overrides the calculated amount and is written to the Break Funding Amount column.
Original Term	Select to apply Original Term to both Fixed and Adjustable Rate Instruments.
Standard Term	Standard Term is the traditional approach used in FTP, which is the original term for Fixed-Rate Instruments and Repricing Terms for Adjustable-Rate Instruments.
Repricing Frequency	Repricing Frequency is the frequency of rate change of a product.
Remaining Term	The remaining Term is the number of months remaining until the instrument matures.

- Availability of Adjustment Methods
  The list of Adjustment Methods depends on the adjustment type that you select: Add-on Rates (including Liquidity Adjustments, Basis Risk Costs, Pricing Incentives, and Other Adjustments), or Breakage Charges.

26.2.1 Availability of Adjustment Methods

The list of Adjustment Methods depends on the adjustment type that you select: Add-on Rates (including Liquidity Adjustments, Basis Risk Costs, Pricing Incentives, and Other Adjustments), or Breakage Charges.

The following table describes the Adjustment Methods available for each of the Adjustment Types.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 26-3 Adjustment Rule Definition - Adjustment Rule Details

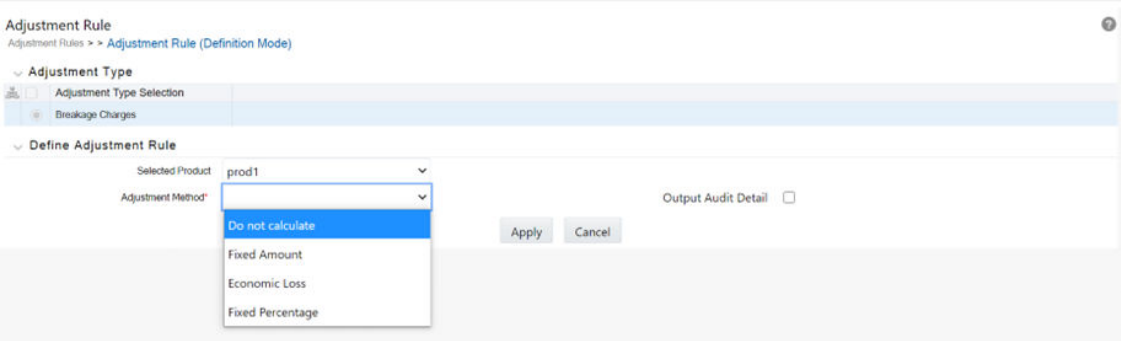


Table 26-2 Availability of Adjustment Methods

Adjustment Method	Adjustment Type: Add-On Rates (Liquidity Adjustments, Basis Risk Costs, Pricing Incentives, and Other Adjustments)	Adjustment Type: Breakage Charges
Do Not Calculate	Yes	Yes
Fixed-Rate	Yes	
Fixed Amount	Yes	Yes
Formula Based Rate	Yes	
Use TP Method from Selected TP Rule	Yes	
Economic Loss		Yes
Fixed Percentage		Yes

> **Note**
>
> If you select **Do not calculate** as the calculation method, no adjustment assumptions will be assigned to the particular product-currency combination. This is a particularly useful option when using node-level assumptions because it allows you to exclude a particular Child from inheriting a Parent's assumption.

- Adjustment Method Parameters
  To define an Adjustment Calculation Method, you must specify one or more parameters, depending on the method. The parameter fields may display a default value, which you can override.

26.2.1.1 Adjustment Method Parameters

To define an Adjustment Calculation Method, you must specify one or more parameters, depending on the method. The parameter fields may display a default value, which you can override.

The following tables display the parameters associated with the Adjustment Methods for different Adjustment Types.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 26-3 Parameters Applicable to the Adjustment Methods for the Add-On Rate Adjustment Types

Adjustment Method	Reference Term	Lookup Method	Term	Multiplier	Rate	Amount	Assignment Date	Interest Rate Code	Formula
Do Not Calculate									
Fixed-Rate	Yes	Yes	Yes	Yes	Yes				
Fixed Amount	Yes	Yes	Yes	Yes		Yes			
Formula Based Rate	Yes	Yes	Yes	Yes			Yes	Yes	Yes
Use TP Method from Selected TP Rule			Yes						

> **Note**
>
> The Add-On Rate Adjustment Types include Liquidity Adjustments, Basis Risk Costs, Pricing Incentives, and Other Adjustments.

Table 26-4 Parameters Applicable to the Adjustment Methods for the Breakage Charge Adjustment Type

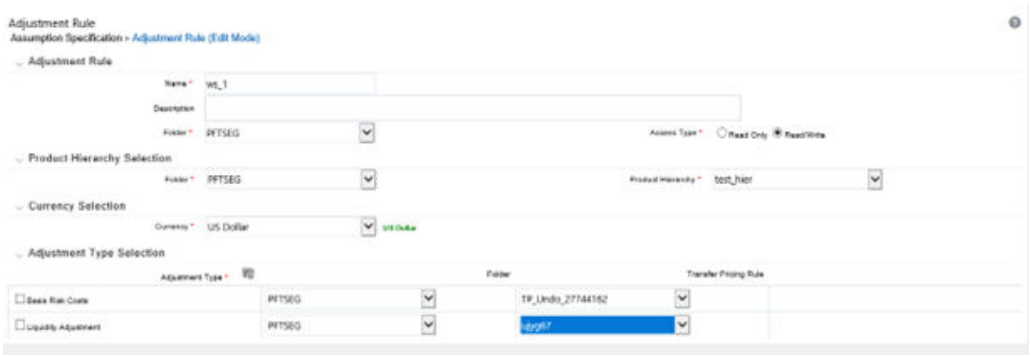
Adjustment Method	Break Funding Amount	Break Funding Rate	Interest Rate Code	Rate Spread	Minimum Charge
Do Not Calculate					
Fixed Amount	Yes				
Economic Loss			Yes	Yes	Yes
Fixed Percentage		Yes			Yes

To define the adjustment method parameters, follow these steps:

1. Perform the basic steps for creating or editing an Adjustment Rule.
   After you have created an Adjustments Rule, you can assign adjustment methodologies to product-currency combinations in either of the following ways:
   - By creating a conditional assumption using conditional logic.
   - Directly on the Adjustment Methodology page, as described here.
2. To define an Adjustment Method, follow these steps:
   a. Navigate to the Adjustments Assumption browser page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 26-4 Adjustment Rule - Edit Mode



b. Select an appropriate Adjustment Type: Liquidity Adjustment, Basis Risk Costs, Pricing Incentives, Other Adjustment, or Breakage Charge by opening the Adjustment Type Selector Window. You can enable one or more Adjustment Types within a single Adjustment Rule and apply more than one Adjustment to a single product.

> **Note**
>
> The product hierarchy refreshes when you change your Adjustment Type selection, but note that all selections made within the rule are saved. For example, when Liquidity Adjustment is selected, the hierarchy displays the status of Liquidity Adjustment mappings within the hierarchy. If you change your Adjustment Type selection to Basis Risk Cost, the hierarchy will refresh and you will see the status of all Basis Risk Cost mappings, and so on.

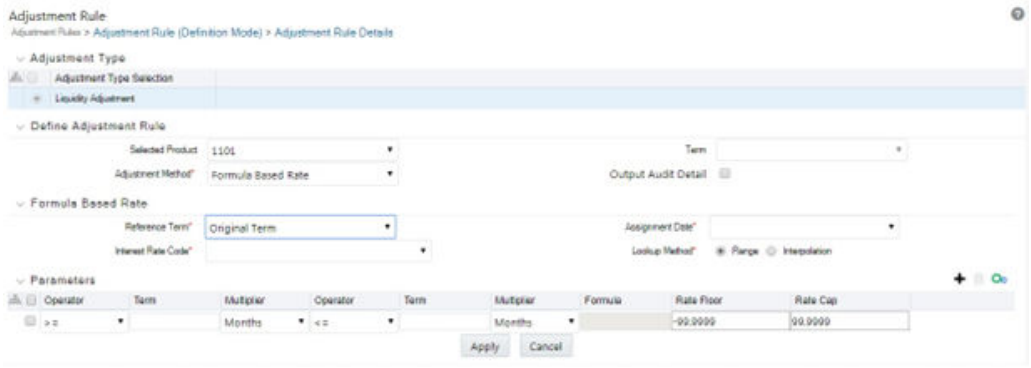
c. Select a **Product Hierarchy**.

d. Specify a **Currency**.
Based on the selected hierarchy, the application displays a list of all the products (for which you can define assumptions).

e. Select the checkbox for one or more products for which you want to define adjustment details.

f. Select an Adjustment Method and enter the appropriate parameters.

Figure 26-5 Adjustment Rule Definition Mode - Adjustment Rule Details



Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The Adjustment Methods available depend on the Selected Adjustment Type. Depending on the Adjustment Type and Adjustment Method combinations selected, certain required and optional parameter fields are displayed. You can update these fields as required.

g. Click **Apply**. If only one product was selected, the Assumption Browser page is displayed. If more than one product was selected on the Assumption Browser page, then each subsequent product in the select list will appear in the Selected Product drop list and each item should be defined appropriately. After completing the assumption details for each selected product, the assumption browser page is displayed. At this point you can:

   - Continue defining assumptions for additional product-currency combinations for the selected Adjustment Type, by repeating the above procedure.
   - Select a new Adjustment Type and continue defining assumption details for the required set of products.
   - Complete the process by clicking **Save**. The new assumptions are saved and the Adjustments Rule summary page is displayed.

> **Note**
>
> **Save** or **Cancel** operation on assumption browser page is only applicable to top half of the page. Changes which are already committed on the assumption browser, like method addition/deletion or conditional assumptions are already committed actions which will remain intact regardless of user clicking save/cancel on assumption browser page.

Oracle Funds Transfer Pricing provides you with the option to copy, in total or selectively, the product assumptions contained within the Adjustments, Transfer Pricing, and Prepayment Rules from one currency to another currency or a set of currencies or from one product to another product or set of products.

26.3 Define the Use TP Method from TP Rule Method

26.3.1 Define the Use TP Method from TP Rule Method

26.3.2 Define the Use TP Method from TP Rule Method

26.3.3 Define the Use TP Method from TP Rule Method

Copyright © 2011, 2026, Oracle and/or its affiliates.

27
Stochastic Rate Indexing

The purpose of the Stochastic Rate Index Rule is to establish a relationship between the risk-free Interest Rate Codes (IRCs) and the other interest rate codes or indices. With this relationship established, you can forecast rates on any instrument tied to an IRC and as the risk-free rates change. The change in the non risk-free interest rates will depending on the established relationship.

The following are the non risk-free interest rate codes:

- Prime
- LIBOR
- Administered rates
- 11th District COFI

The Stochastic Rate Index Rule is used only in stochastic processing. For more information, see the Defining Stochastic Rate Indexes section.

The procedure for working with and managing the Stochastic Rate Index Rule is similar to the other Oracle Asset Liability Management Business Rules. It includes the following steps:

- Search Stochastic Rate Index Rule. For more information, see the Searching for Rules section.
- Defining Stochastic Rate Index rules
- View and Edit Stochastic Rate Index Rule. For more information, see the View and Edit the Rule section.
- Copy Stochastic Rate Index Rule. For more information, see the Copy the Rule section
- Delete Stochastic Rate Index Rule. For more information, see the Delete the Rule section
- Defining Stochastic Rate Indexes

27.1 Defining Stochastic Rate Indexes

A formula must be defined for each interest rate index tied to an instrument. The formula must be in the following form:

Figure 27-1 Stochastic Rate Index Formula

$$Index\cdot Rate_{term.m}\cdot=\cdot K_1\cdot Risk\cdot Free\cdot Rate_1^{x1}\cdot+\cdot K_2\cdot Risk\cdot Free\cdot Rate_2^{x2}\cdot+\cdot\ldots K_8\cdot Risk\cdot Free\cdot Rate_8^{x8}\cdot+\cdot Spread$$

To create a formula, select up to eight terms (elements) from the risk-free curve, each multiplied by a user-defined coefficient, and raised to the power of a user-defined exponent. Also, you can add a constant spread to the formula. It is not necessary to define any assumptions for the risk-free curve. Any definition for this curve is ignored and does not affect processing.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Each of the elements consists of:

- **A coefficient**: A multiplier to weigh each term selection.
- **An exponent**: An exponent to allow for polynomial curve-fitting.
- **A term selection**: A selection of rates associated with a term from the risk-free curve.

These elements define a different rate forecast generated for each instrument, with a given IRC. The definition of rate indexes is part of the **Create Stochastic Rate Index Rule** process in which rate indexes are defined for currency-valuation curve combinations. When you click **Save** in the **Create Stochastic Rate IndexRule** process, the rule is saved and the **Stochastic Rate Index Rule** summary page is displayed. However, the rate indexes have not yet been defined for any of the currency-valuation curve combinations. Define the rate indexes for currency-valuation curve combinations before clicking **Save**.

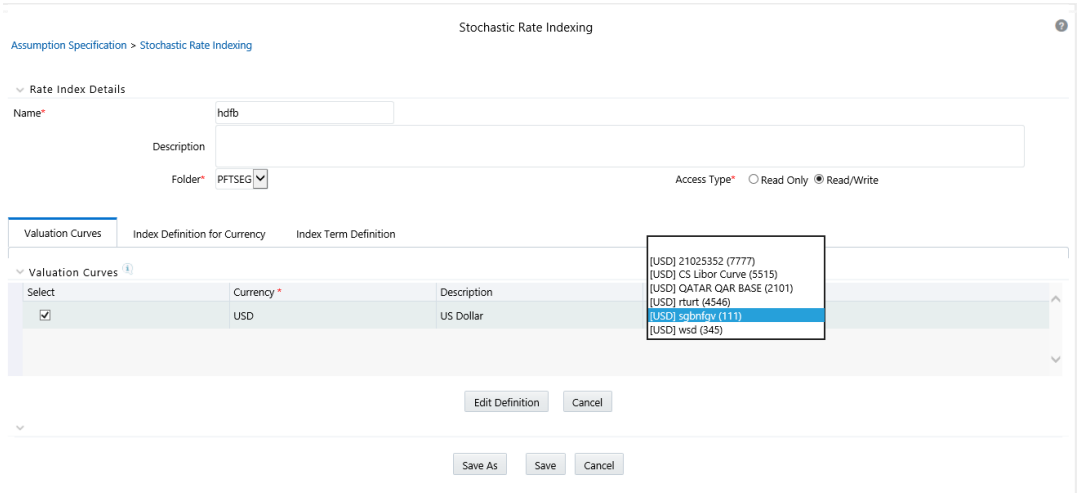
Prerequisites

- Decide on historical rate information and manage by creating Interest Rate Codes.
- Perform the steps to create or edit a Stochastic Rate Index rule.

To create a Stochastic Rate Index rule, follow these steps:

1. Navigate to the **Stochastic Rate Index Valuation Curve Assignment** page.

Figure 27-2 Stochastic Rate Indexing



2. Enter the following information as tabulated.
   Rate Index Details Window

   Form fields to Define Stochastic Rate Indexes

Copyright © 2011, 2026, Oracle and/or its affiliates.

Term	Description
Valuation Curve	The Valuation curve is used to calculate the future rates of Indexes (IRCs) defined in a Stochastic Rate Index rule. Assign a Valuation curve for each currency during the creating process of the Stochastic Rate Index rule. The Valuation curve and the indexed rate curves are derived from the same Referenced Currency. For example, use the US Treasury Yield Curve as the Valuation curve to calculate the forward rates of any US dollar-based Interest Rate Code.
Coefficient	A multiplier to weigh each term selection.
Exponent	An exponent to allow for polynomial curve-fitting.
Index Term	An Interest Rate Code is made up of one or many term points that denote a particular interest rate yield curve structure. OFS ALM 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 the 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. **Figure 27-3 Stochastic Rate Indexing Formula** $$\text{Index Rate}_t = \sum_{n=8}^{1} [\text{Coefficient}_n \times (\text{RFR}_T)^{\text{Exponent}_n}] + \text{Spread}_n$$ 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 a formula, select up to eight-term points (elements) from the risk-free curve. Each must be multiplied by a user-defined coefficient and raised to the power of a user-defined exponent. Also, add a constant spread for each of the term points used in the formula.

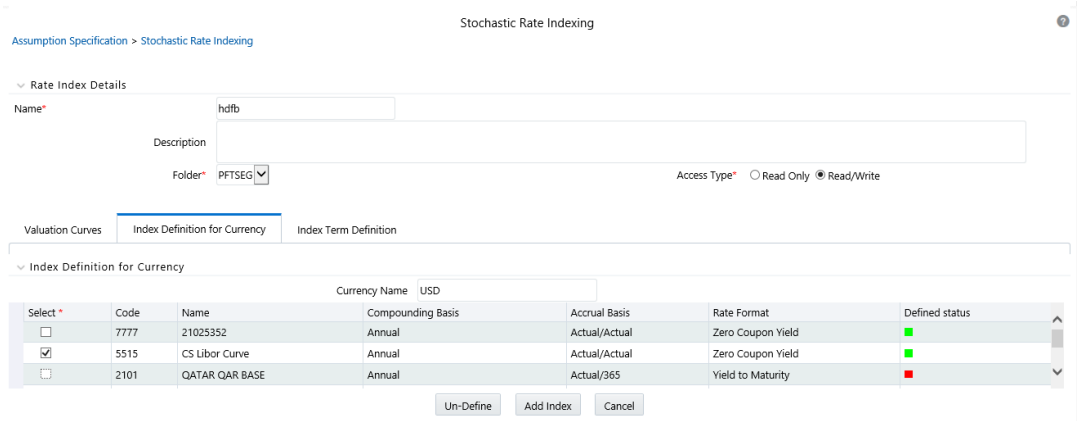
3. Select **Currency**.
4. Navigate to the **Valuation Curve** tab and select a **Valuation Curve** for the selected currency .

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> Only a single Valuation Curve can be associated with a particular currency. For example, if the Valuation Curve for US Dollars is US Treasury Curve, all US Dollar indexes will be associated with the US Treasury curve.
>
> Select a risk-free 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 any curve as the Valuation Curve.

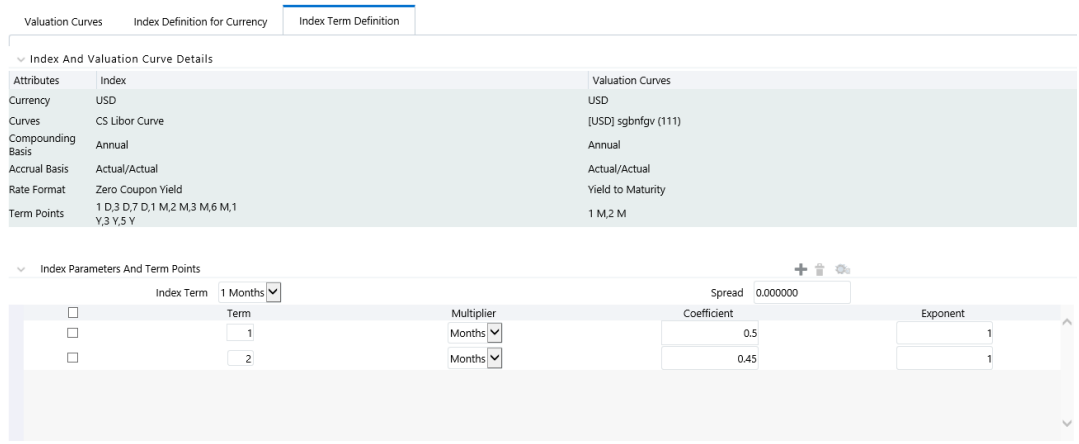
5. Select **Apply**.
6. Navigate to the **Index Definition for Currency** tab and select the Index using **Add Index**.

Figure 27-4 Stochastic Rate Indexing



7. Navigate to the **Index Term Definition** tab. The attributes of both the valuation curve and the selected IRC, are listed in Table 1. This information can be used as a reference to define the terms.
8. Add **Index Term Definitions**. The **Add Index Term Definition** Page is displayed. The general attributes of both the Valuation Curve and the selected IRC, are displayed. This information can be used as a reference when you define the terms.
   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.

Figure 27-5 Stochastic Rate Indexing



Copyright © 2011, 2026, Oracle and/or its affiliates.

9. Select the **Index Term**.
   Not all IRCs have Term Points defined. To successfully define an Index, you must define at least one of its terms. Else define one, many, or all of the Index Terms. The selection of the Index Term is limited to the standard Term Points as defined in the IRC definition.
10. Enter a **Spread** for the **Index Terms**, if required. A Spread is a constant percentage added to the rate produced as a result of the Monte Carlo calculations, multiplied with the defined coefficient, and raised 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.
13. Enter an **exponent** for the element.
14. Repeat the last four steps for a maximum of seven more elements for each term.
15. Click **Apply** when indexing is completed for all the required term points.
    The **Stochastic Rate Index Valuation Curve** page is displayed. You can navigate to the **Index Definition** tab and continue defining rate index relationships for the existing currency selection or switch currencies, define the valuation curve for the new currency and proceed with Rate Index definitions.

> **Note**
>
> During rule creation, on the **Index Definition forCurrency** tab, the undefined IRCs are displayed in the top half of the window and the defined IRCs are displayed in the bottom half of the window. After saving the rule and re-entering in Edit mode, you can see the status reflected for all defined (green) and undefined (red) IRCs in the top half of the window. To Edit any Rate Index relationships, select the appropriate IRC, and again select **Add Index** and proceed with any changes.

16. Click **Save**.
17. The **Stochastic Rate Index Rule Summary** page is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

28

Alternate Rate Output Mapping Rules

In Oracle Funds Transfer Pricing, you either can output Transfer Pricing Results to the default columns of the application, or to the seeded or user-defined alternate columns selected using the Alternate Rate Output Mapping Rule.

Both the Standard Transfer Pricing Process and Stochastic Transfer Pricing Process reference the Alternate Rate Output Mapping Rule to Output Transfer Rate, Option Cost, and Adjustment Calculation Results for each Instrument Record.

The procedure for working with and managing the Alternate Rate Output Mapping Rule is similar to that of other Oracle Funds Transfer Pricing Business Rules. It includes the following steps:

- Searching for Alternate Rate Output Mapping Rules
- Creating Alternate Rate Output Mapping Rules
- Viewing and Editing Alternate Rate Output Mapping Rules
- Copying Alternate Rate Output Mapping Rules
- Deleting Alternate Rate Output Mapping Rules

Before creating Alternate Rate Output Mapping Rules, you need to register any user-defined columns that you have added as Alternate Output Columns for instrument tables. Note that a full set of Alternate Target Columns is seeded with each instrument table, so it is not a requirement to create and register User-Defined Columns. You can either utilize the seeded alternate columns or create your own custom alternate columns.

This chapter describes the procedure to output Transfer Pricing results to the seeded or User-Defined Alternate Columns instead of Default Columns of the application.

Alternate Rate Output Mapping Rules allow you to map Transfer Pricing results to alternate or User-Defined Columns rather than to the Standard Output Columns. Alternate Rate Output Mapping rules are optional components of a Transfer Pricing process. If these rules are excluded from a process, then results are written to the standard default columns on the instrument tables. If an Alternate Rate Mapping table is included then outputs will be written based on target columns specified by the user. This functionality allows users to calculate and output more than one Transfer Rate, Option Cost, or TP Add-on Rate for each Instrument Record.

- Create Alternate Rate Output Mapping Rules
  You should register the Alternate Output Columns for Account Tables before creating the rules.
- Registering Alternate Output Columns for Account Tables
  It is possible to add User-Defined Columns to your instrument tables and to designate certain columns as target columns for Alternate Rate Output.

Copyright © 2011, 2026, Oracle and/or its affiliates.

28.1 Create Alternate Rate Output Mapping Rules

You should register the Alternate Output Columns for Account Tables before creating the rules.

You create an Alternate Rate Output Mapping Rule to select the alternate columns to Output Transfer Rate, Option Cost, and Adjustment Calculation Results for each instrument record in an account table for a Transfer Pricing Process Run.

1. Navigate to the Alternate Rate Output Mapping Rule Summary page.
2. Complete the standard steps for this procedure.

- Additional Steps to Create Alternate Rate Output Mapping Rule
  In addition to the Standard Steps for creating rules, the procedure for creating Alternate Rate Output Mapping Rules involves the following extra steps:

28.1.1 Additional Steps to Create Alternate Rate Output Mapping Rule

In addition to the Standard Steps for creating rules, the procedure for creating Alternate Rate Output Mapping Rules involves the following extra steps:

1. Select the instrument table for which you want to output Transfer Pricing results into Alternate Columns.
2. (Optional) Select an Alternate Column Mapping for the following result types:
   - Transfer Rates: Transfer Rate, Transfer Rate Charge Credit, Matched Spread, and Remaining Term Transfer Rate. When selecting an alternate Transfer Rate Column, you must select an alternate Matched Spread Column.
   - Option Costs: Historic Option Adjusted Spread, Historic Static Spread, Current Option Adjusted Spread, and Current Static Spread.

   Option Cost Output Columns are displayed only for tables registered with the Option Cost classification.
   - **Adjustments**: Liquidity Premium Rate, Liquidity Premium Rate Charge Credit, Liquidity Premium Amount, Basis Risk Cost Rate, Basis Risk Cost Rate Charge Credit, Basis Risk Cost Amount, Pricing Incentive Rate, Pricing Incentive Rate Charge Credit, Pricing Incentive Amount, Other Adjustment Rate, Other Adjustment Rate Charge Credit, Other Adjustment Amount. If you are using Alternate Rate Output, you should define Alternate Columns for all Transfer Rate and Adjustment Rate Columns.
   - **Break Funding Charges**: If the source table is Break Funding Charges, then the following Alternate Output Columns can additionally be defined. Break Funding Market Value, Break Funding Amount, Break Funding Rate, and Break Funding Amount Change.
   - **Economic Value Outputs**: The Economic Value drop-down contains the following list of values with the corresponding linked columns: Economic Value All in TP Rate, Economic Value Basis Risk Cost Rate, Economic Value Liquidity Premium Rate, Economic Value Other Adjustment Alternate Rate, Economic Value Other Adjustment Rate, Economic Value Pricing Incentive Rate, and Economic Value Transfer Rate.
   - (Optional) **Deselect the Apply Mappings to All Tables option**: The default setting of the Apply Mappings to All Tables option is selected. This functionality lets you apply Alternate Column Mappings from the current page to all other Instrument Tables in which the selected Result Columns are available and replace any previous selections in the other instrument tables. If a result type exists in the target table, but is not displayed on the current page (for example, when the target table is classified for

Copyright © 2011, 2026, Oracle and/or its affiliates.

Option Cost, but the current source table does not have the Option Cost classification), the rule maps results to default columns in the target table. For example, it would map to Historic Option Adjusted Spread Alternate Column, Historic Static Spread Alternate Column, and so on). If you deselect the Apply Mappings to All Tables option, the rule saves mappings to the Default Columns on any table for which you have not explicitly selected Alternate Output Columns.

28.2 Registering Alternate Output Columns for Account Tables

It is possible to add User-Defined Columns to your instrument tables and to designate certain columns as target columns for Alternate Rate Output.

The following steps will allow you to register these columns for use within the application and will allow you to select the columns from within the Alternate Rate Output Mapping Rule screen.

1. Create the new user-defined column(s) in the Erwin Model and define the new User-Defined Property (UDP) for the column(s).
2. For Alternate Rate Output Mapping, the following are the applicable UDP's:
   - Transfer Pricing Output (80)
   - Option Cost Output (81)
   - Liquidity_rate_column (95)
   - Liquidity_amount_column (96)
   - Basis_rate_column (97)
   - Basis_amount_column (98)
   - Pricing_rate_column (99)
   - Pricing_amount_column (100)
   - Other Adjustment Spread Output (82)
   - Other Adjustment Amount Output (83)
   - Economic Value Output (86)
   - Break Funding Market Value (91)
   - Break Funding Amount (90)
   - Break Funding Rate (92)
   - Break Funding Amount Charge (93)
   - Transfer Pricing Charge Credit (101)
   - Liquidity Premium Charge Credit (102)
   - Basis Risk Charge Credit (103)
   - Pricing Incentive Charge Credit (104)
   - Other Adjustment Charge Credit (105)

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> When adding or modifying an instrument table column within Erwin, the UDP properties tab lists all the applicable properties available at a column level. The above UDP's will correspond to the Alternate Rate Output Column types noted above. You need to specify the value of the relevant property as YES (in CAPS) to enable display in the appropriate section of the Alternate Rate Output Mapping screen.

3. Upload the modified Data Model. After the saved model is uploaded, and the table classification procedure re-executed, the **Alternate Rate Output Mapping** screen displays the new columns.
4. Depending on the selected UDP's, the new column(s) appear in the appropriate drop list within the **Alternate Rate Output Mapping Definition** screen.
   After you make appropriate Column User-Defined Property Assignment to the alternate columns, they become available in the Alternate Rate Output Mapping rule under the appropriate result types. Selecting Alternate Output Columns for each Transfer Pricing Column is a one-time setup process. However, the application lets you modify the Alternate Output Columns setup, if necessary.

> **Note**
>
> For further information on the Table Column User-Defined Property Assignment step of the table registration process, see the OFS Analytical Applications Infrastructure Installation and Configuration Guide and OFS Data Model Utilities User Guide.

Copyright © 2011, 2026, Oracle and/or its affiliates.

29

Propagation Patterns

The Propagation Pattern allows you to define source tables and Lookup terms required for propagating Transfer Rates and Option Costs for any applicable instrument table from a prior period.

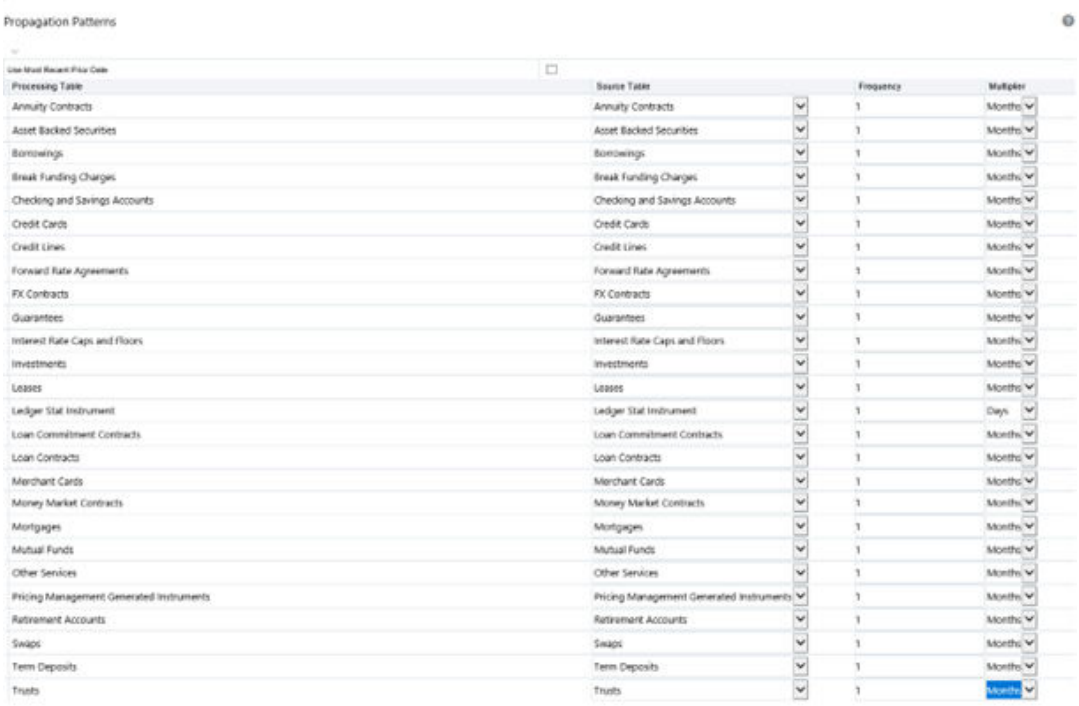
- Defining the Propagation Pattern
  To define the Propagation Pattern:
- Propagating Transfer Pricing Results
  Depending on your requirements, you can choose to propagate Transfer Rates, Adjustment Rates, or Option Costs, by selecting the appropriate Propagation Processing Option in the Transfer Pricing Process.

29.1 Defining the Propagation Pattern

To define the Propagation Pattern:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Maintenance**, select **Patterns**, and then select **Propagation Patterns** to display the Propagation Patterns summary page.

Figure 29-1 Propagation Patters summary



2. Enter or select the following:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 29-1 Fields and Descriptions from the Propagation Patterns summary page

Term	Description
Processing Table	Instrument tables that are 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.
Frequency	A numeric value multiplied with a Multiplier to calculate the Historical Lag reference date for rate Lookups.
Multiplier	The unit value of the Frequency.
Use Most Recent Prior Date	If Use Most Recent Prior Date option is not on, then As-of-Date – Lookback term is calculated. If Use Most Recent Prior Date option is on, the nearest prior As-of-Date available in the instrument record is used for calculation. The Lookback term from the Propagation UI is ignored.

3. Select the **Source Table** that needs to be associated with each 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 Instrument table) or a separate table (if you store historical records in separate Instrument tables).

4. Specify the **Historical Lag** between the Processing and Source Tables.
   - Select the **Frequency**.
   - Select the **Multiplier**.

> **Note**
>
> The prior period SourceID for each Source Table is defined with the current As-of-Date. For instance, if you transfer price monthly, you should specify the historical lag between the Processing and Source Tables as one month. Alternatively, select the “Use Nearest Prior Date" option to have the system automatically determine the prior date.

5. Click **Save**.
   The Propagation Pattern assumptions that you defined are saved.
6. Click **Reset** to restore default values. This selection will set the Processing and Source Tables equal to each other and will set the Term and Frequency equal to 1 Month, for all rows.

- Loan Commitment Propagation
  When a Loan Commitment is originated, it has a commitment number rather than an account number. To support the propagation of TP Results for Loan Commitment Contracts, users can choose to match current and prior records based on the commitment number rather than the ID Number. This capability allows users to propagate from month-to-month or day-to-day within the Loan Commitment Table and after the loan is booked,

Copyright © 2011, 2026, Oracle and/or its affiliates.

from the Loan Commitment Table to one of the loan tables such as Mortgages or Loan Contracts.

29.1.1 Loan Commitment Propagation

When a Loan Commitment is originated, it has a commitment number rather than an account number. To support the propagation of TP Results for Loan Commitment Contracts, users can choose to match current and prior records based on the commitment number rather than the ID Number. This capability allows users to propagate from month-to-month or day-to-day within the Loan Commitment Table and after the loan is booked, from the Loan Commitment Table to one of the loan tables such as Mortgages or Loan Contracts.

You can enable or disable Loan Commitment Propagation. If this option is enabled, then Loan Commitment Propagation will run after Standard Propagation.

To define a Loan Commitment Propagation:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Maintenance**, select **Patterns**, and then select **Propagation Pattern**.

Figure 29-2 Loan Commitment Propagation



Table 29-2 Fields and Descriptions from the Loan Commitment Propagation Patterns summary page

Term	Description
Processing Table	This list includes the seeded instrument tables that hold loan contracts.
Source Table	This list includes any Instrument tables that are classified as a Loan Commitment Table, for example, Loan Commitment Contracts.
Target Column to Match	This list contains the column to use for matching the current record and prior record. The current option is Commitment Number (COMMITMENT_NBR). The propagation process will match the value in the ID_NUMBER column from the Loan Commitments Table with the COMMITMENT_NBR value from the Loan Contracts or Mortgages Table.
Frequency	A numeric value multiplied with a Multiplier to calculate the Historical Lag reference date for rate Lookups.
Multiplier	The unit value of the Frequency.

Select the checkbox option for Loan Commitment Propagation (if applicable).

2. Select the **Source Table** that needs to be associated with each **Processing Table**.
3. Select the **Target Column** to **Match as Commitment Number**.
4. Specify the **Historical Lag** between the processing and source tables.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Select the **Frequency**.
- Select the **Multiplier**.

5. Click **Save**.

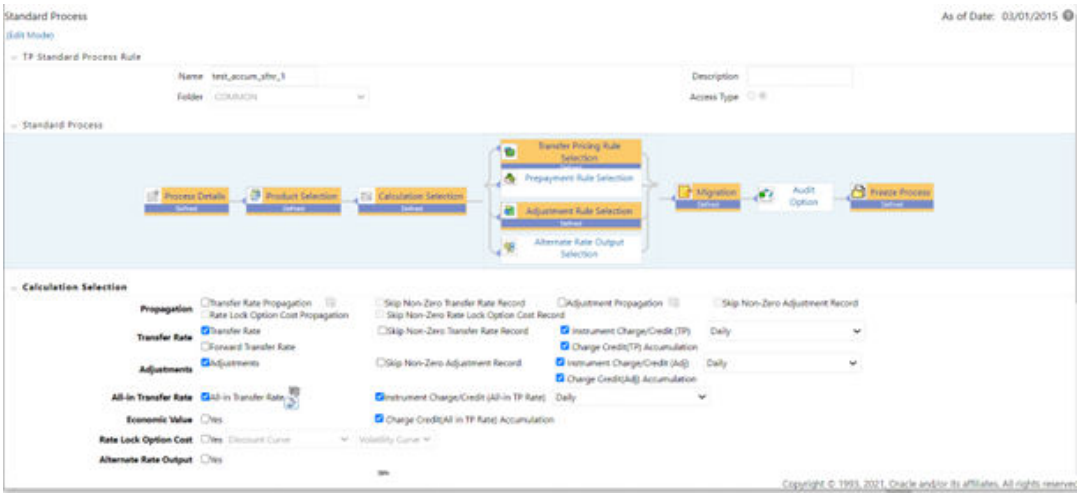
29.2 Propagating Transfer Pricing Results

Depending on your requirements, you can choose to propagate Transfer Rates, Adjustment Rates, or Option Costs, by selecting the appropriate Propagation Processing Option in the Transfer Pricing Process.

The propagate Transfer Pricing results, follow these steps:

1. Create or update a Transfer Pricing Rule.
2. Navigate to the Transfer Pricing Process Calculation Selection block.

Figure 29-3 Transfer Pricing Rule - Calculation Selection



3. Select the propagation parameters:

- From a Standard Transfer Pricing Process, select Transfer Rate and select the Transfer Rate Propagation option and/or Adjustment Propagation option. Selecting Transfer Rate Propagation updates all term-related instrument records, which have an instrument-level history for a prior period with the Transfer Rate that applied in that prior period. If Adjustment Propagation is selected, then all Adjustment outputs (except breakage charges) including rates and amounts will be propagated.

> **Note**
>
> If you have pre-populated some Transfer Rates or Add-on Rates before running propagation and you would like the propagation process to skip these records, then select the Skip Non-Zero Transfer Rate Record option and or Skip Non-Zero Adjustment Record option.

- From a Stochastic Transfer Pricing Process, select Option Costs and select the Option Cost Propagation option and/or Adjustment Propagation option. Selecting Option Cost Propagation updates all term-related instrument records, which have an instrument-level history for a prior period with the Option Cost Data that applied in that Prior

Copyright © 2011, 2026, Oracle and/or its affiliates.

Period. If Adjustment Propagation is selected, then all of the Adjustment Outputs (except breakage charges) including rates and amounts will be propagated.

> **Note**
>
> If you have pre-populated some option costs and you would like to propagation process to skip these records, then select the Skip Non-Zero Option Cost Record option.

For more information, see the Transfer Pricing Process.

> **Note**
>
> When a table is updated using a propagation pattern, an instrument record must satisfy the following criteria to receive a Transfer Rate, Adjustment Rate, or Option Cost.
>
> - It must be an instrument that exists in both the Target (processing) Table (with the current As-of-Date) and the Source Table (with the prior period based on a matching ID_NUMBER).
> - 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 <> 0 in Target table) instrument with Target Last Repricing Date <= Prior Period As-of-Date. In other words, it must be an Adjustable-Rate Instrument that has not been Repriced since the prior period.
>   The matched spread is also migrated from the prior period record and not recomputed from the Transfer Rate and Current Rate on the Target Table Record.
> - For Adjustment Rate Propagation, all rates are propagated regardless of fixed or adjustable type. If additional logic is required to control the Propagation of Adjustment Rates, data filters should be used to specify the conditions.

Copyright © 2011, 2026, Oracle and/or its affiliates.

30

Transfer Pricing Process

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

Transfer Pricing Processes are split into two types:

- Standard: Allows you to calculate Transfer Rates and Adjustments.
- Stochastic: Allows you to calculate Option Costs.

The Transfer Pricing Process allows you to perform the following tasks:

- Determine the data that you want to process (Product Selection block).
- Submit to the Transfer Pricing Engine the Transfer Pricing, Prepayment, and Adjustments Assumptions you want to process (Rule Selection block).
- Specify to the Transfer Pricing engine whether you want to generate Transfer Rates, TP Rate Adjustments, Economic Value, or Option Costs (Calculation Selection).
- Specify to the Transfer Pricing Engine whether you want to calculate or propagate Transfer Pricing Results (Calculation Selection).
- Specify to the Transfer Pricing Engine the alternate columns in which to output Transfer Rate, Option Cost, Economic Value, and Adjustment Calculation Results for each instrument record in an account table for a Transfer Pricing Process Run (Alternate Rate Output Selection block).
- Calculate and migrate charges and credits for funds provided or used as well as rate lock option costs or breakage charges for a combination of dimensions to the Management Ledger Table (Migration block).
- Enable the output of detailed cash flows for audit purposes (Audit Options block).
- Formulate and execute the transfer pricing request and generate results (Transfer Pricing Process summary page).

For more information, see Define and Execute a Transfer Pricing Process.

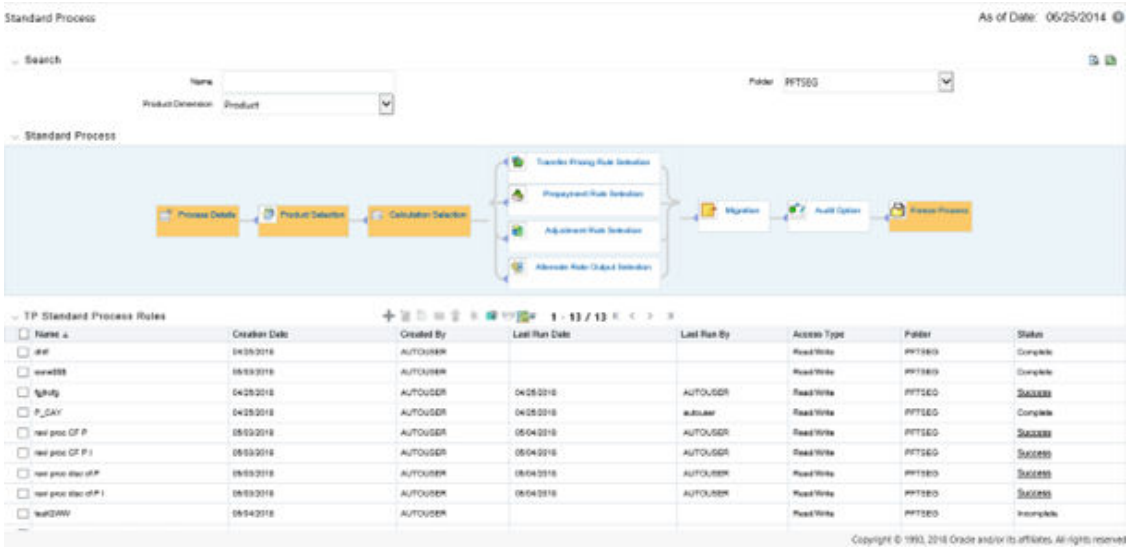
The procedure for working with and managing the Transfer Pricing Process is similar to that of other Oracle Funds Transfer Pricing business rules. It includes the following steps:

- Searching for Transfer Pricing Processes
- Creating Transfer Pricing Process Rules
- Viewing and Editing Transfer Pricing Processes
- Copying Transfer Pricing Processes
- Deleting Transfer Pricing Process Rules

Transfer Pricing Processes are executed from the Transfer Pricing Process summary page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-1 Transfer Pricing - Standard Process



- Create a Standard Transfer Pricing Process
  Create a Standard Transfer Pricing Process:
- Creating a Stochastic Transfer Pricing Process
  Create a Stochastic Transfer Pricing Process:
- Execute a Transfer Pricing Process
  You execute a Transfer Pricing Process:

30.1 Create a Standard Transfer Pricing Process

Create a Standard Transfer Pricing Process:

- To define and execute Transfer Pricing Processing Requests.
- To calculate Transfer Rates, TP Rate Adjustments (TP Add-on Rates), and related charge or credit amounts.
- To propagate Transfer Rates or Adjustment Rates for any Applicable Instrument Table from a prior period.
- To migrate charges or credits, for funds provided or used, to the Management Ledger Table.
- To Calculate the All-in Transfer Rate.
- To Calculate Economic Value of the Portfolio.
- To calculate and/or migrate Rate Lock Option Costs.
- To calculate and/or migrate Breakage Charges.
- To select alternate columns to Output Transfer Rate or Adjustment Calculation Results for each record in an instrument table for a Transfer Pricing Process Run.
- To generate Audit Trail Output for Assumption Rule or for All Data.

The prerequisites for defining and executing the Standard Transfer Pricing Processing Requests are:

- Performing basic steps for creating or editing a Transfer Pricing Rule

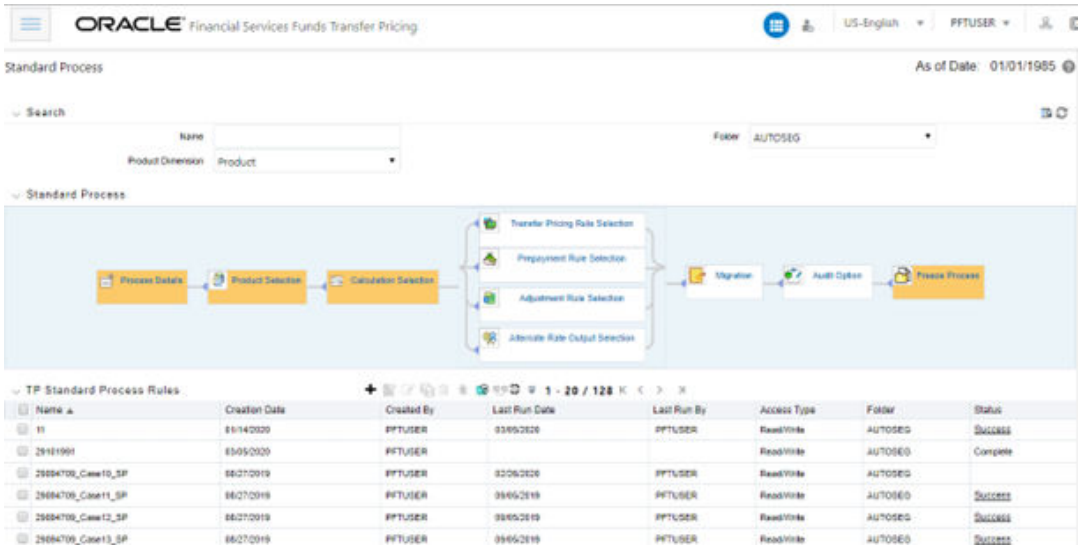
Copyright © 2011, 2026, Oracle and/or its affiliates.

- Performing basic steps for creating or editing an Adjustment Rule

To create and execute the Standard Transfer Pricing Process:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Processing**, and then select **Standard Process**.

Figure 30-2 FTP Processing - Standard Mode summary page



The following table describes the key terms used for this procedure:

Table 30-1 Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
Product Hierarchy	Displays the selected product hierarchy. The Product Hierarchy allows you to select Product Members to include in the Process. Note, you have two methods for selecting data that will be included in the process. You can select Parents / Children from a hierarchy or you can use Source selection.
Source	Allows you to select one or more source tables to include in your process. If sources are selected directly then the Product Hierarchy will be disabled and selections made within the hierarchy will be lost. Users can utilize one method or the other to define data that will be included in the process, but not both methods at the same time.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-1 (Cont.) Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
Filter	Filters allow you to restrict your data selection based on any attribute that exists within an instrument table. You define filters under Common Object Maintenance and reference your filter within the Product Selection block of your Process. The supported Filter Types are Attribute Filter, Data Filter, Hierarchy Filter, and Group Filter.
Transfer Pricing rule	This LOV allows you to select a Transfer Pricing rule. The Transfer Pricing rule provides the application with assumptions that are applied to products to calculate their transfer rates and option costs.
Prepayment Rule	This LOV allows you to select a Prepayment rule. The Prepayment rule provides the application with prepayment assumptions that you want to apply to products in conjunction with cash flow transfer pricing methods.
Adjustments Rule	This LOV allows you to select an Adjustments rule. The Adjustments rule lets you apply transfer pricing add-on rates to an instrument record for purposes of determining specific charge and credit amounts.
Alternate Rate Output Mapping Rule	This LOV allows you to select an Alternate Rate Output Mapping rule. The Alternate Rate Output Mapping rule lets you select the alternate columns to output transfer rate, and adjustment calculation results for each instrument record in an account table for a standard transfer pricing process run. This functionality allows you to output more than one transfer rate, or adjustment calculation result for each record in the instrument table through multiple transfer pricing process runs.
Calculation Method	Within the Calculation Selection block, there are two calculation options: Standard or Remaining Term. Standard is the most commonly used method and applies traditional transfer pricing logic. The remaining Term allows you to assign Transfer Rates based on current period rates.
Interpolation Method	A calculation selection parameter, the Interpolation Method allows users to decide between Linear, Cubic Spline, or Quartic Spline interpolation methods. This selection affects how to rate lookups happen for terms that fall between anchor points on your Interest Rate Curves.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-1 (Cont.) Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
Charge Credit Method	A migration block parameter, the Charge Credit Method selection determines how the Ledger Level TP Charge or Credit is calculated during the migration process. There are two options available: • **Instrument Level**: With this method, the instrument level charges and credits that were posted during the Transfer Pricing process are added and posted to the Ledger table, together with the weighted trarate. • **Ledger Level**: with this method, the balance of the instrument (average, ending, or other) to arrive at a weighted average transfer rate weights the instrument level transfer rates. The weighted average transfer rate is migrated to the Ledger table and multiplied by the corresponding Ledger Balance (average or ending) to arrive at the charge or credit amount.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-1 (Cont.) Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
Charge/Credit Accrual Basis	Select the Charge/Credit Accrual Basis on the Transfer Pricing Process Migration block or, define the Accrual Basis as an attribute for each Product dimension member in the Dimension Management – Members screen. In case no selection is made, a default Accrual Factor of Actual/Actual should be applied. Charge/Credit Accrual Basis denotes the basis on which the Ledger level interest accrual is calculated. You need to select the accrual factor to be applied when calculating the Ledger Level cost of funds. For instrument-level charges and credits, the accrual basis from the instrument record is always used. When the Ledger Level charge/credit method is selected, the cost of funds is calculated using the following formula: Cost of Funds = Balance x Assigned Transfer Rate x Charge/Credit Accrual Factor **Note** The type of balance is determined by the TP Charge Credit Balance selection made within TP Application Preferences. Choices include Average Balance, Ending Balance, or Other Balance. Oracle Funds Transfer Pricing offers the following accrual basis options: • **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 the number of days in the month divided by 360 days. • **Actual/Actual**: Applies the accrual basis calculation of the number of days in the month divided by the number of days in the year. • **30/365**: Applies the accrual basis calculation of 30 days divided by 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 the number of days in the month divided by 365 days. • **Business/252**: Applies the accrual basis calculation of the number of business days in the month divided by 252 days. A Holiday

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-1 (Cont.) Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
	calendar selection is required if the business/252 accrual basis is selected. If the holiday calendar is not selected, the engine considers Accrual type ACT/ACT as a default for calculation.
Accrual Type	A Calculation Selection Parameter and Migration Block Parameter. Users have the option to choose between Monthly and Daily charge/credit accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.
Currency Output	A migration block parameter, allows you to select the output currency. Oracle Funds Transfer Pricing offers you the following currency output options: • Entered and Functional Currency • Functional Currency Only For example, a bank's loan may have Yen as entered currency. However, the bank might use the US dollar to display its consolidated annual results. In this case, the US dollar is the functional currency. In other words, the currency in which an organization keeps its books is its functional currency.
Migration Dimensions	Dimensions for which you want to migrate charges or credits, for funds provided or used, to the Management Ledger Table. Oracle Funds Transfer Pricing provides you with a Shuttle Control component to specify migration dimensions.

Copyright © 2011, 2026, Oracle and/or its affiliates.

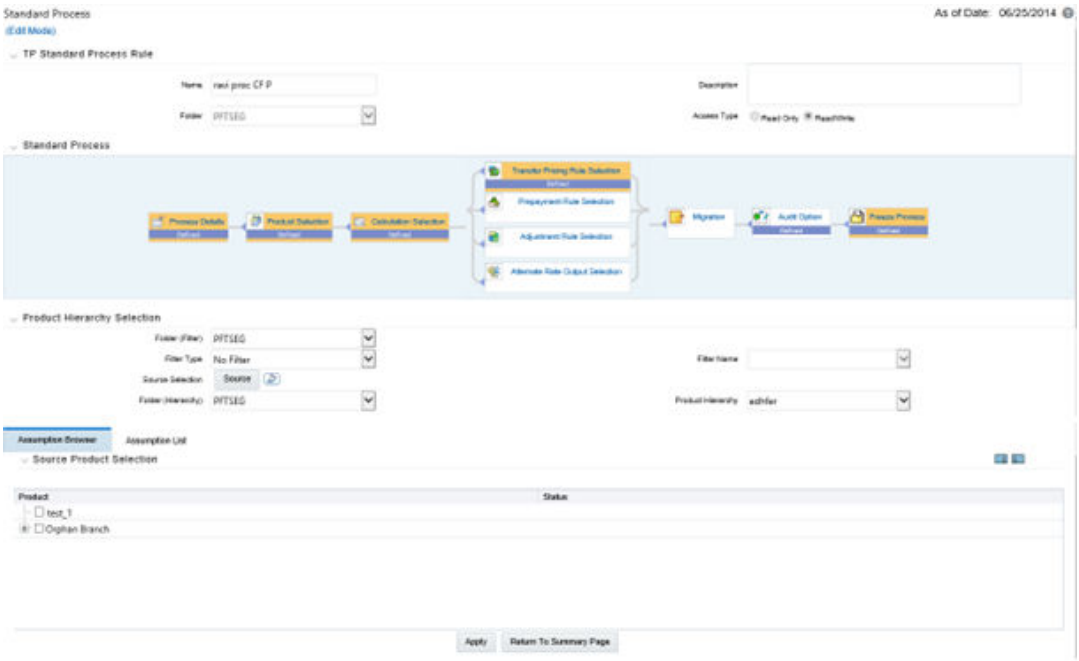
Table 30-1 (Cont.) Fields and Descriptions from the FTP Processing – Standard Mode summary page

Term	Description
Available Dimensions	One of the two Shuttle Control windows, it contains the names of the dimensions available, in addition to the selected Product, for inclusion during the migration process. The dimensions available are: • Common Chart of Account • Organizational Unit • GL Account • Product • Legal Entity **Note** Only the Management Table processing key dimensions are available for inclusion during the migration process.
Selected Dimensions	One of the two Shuttle Control windows, it contains the names of the dimensions, in addition to the selected Product dimension that has already been selected for migration.
Audit Options	Within the Audit Options block, you have the option to output detailed cash flows. Select the checkbox to enable this option and define the number of instrument records for which you would like to output details. Note, the Administrator can limit the maximum number of instruments for which you can compute cash flows because this can be a time-consuming process. In any case, the maximum allowable value is 10,000 instruments. In addition, the **Enable FTP Audit Trail Output** option allows you to select the relevant option to either reading the Audit Preference from the TP or Adjustment assumption rule or generating Audit Trail for all data. If you select the check box **Enable FTP Audit Trail Output**, then it is mandatory to select one of the options for the Audit Trail. When you select the **Enable Audit Trail Output** option, by default **Read Audit Preferences from Assumption Rule** is selected.
Freeze Process	The freeze process block allows you to finalize the assumptions made in the Process definition flow or to clear all assumptions.

2. Complete standard steps for this procedure.
3. Additional Steps to Specify Standard Transfer Pricing Process Parameters.
4. Product Selection Block:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-3 Standard Process Edit Mode



a. Select a Product Hierarchy and select the nodes from the hierarchy corresponding to the data you want to include in the process, OR

b. Alternatively, select the Source Selection button, and select the instrument tables corresponding to the data you want to include in the process.

c. Select a Filter (optional), to constrain the data to be included in the process. Filters work as a secondary constraint, applied after the data set is determined based on Product Hierarchy member selection or Source Selection.

> **Tip**
>
> Before using the product hierarchy approach for selecting data to include in your process, there is a procedure that must be run (“PRODUCT TO INSTRUMENT MAPPING”). This procedure can be executed from the Batch Scheduler – Run - interface.

The purpose of the Product to Instrument mapping procedure is to scan all instrument tables (FSI_D_xxx) and populate the mapping table (FSI_M_PROD_INST_TABLE_MAP) with a listing of the product dimension members that exist within each instrument table. When you select Products (parents or children) within an ALM or FTP process definition, the process refers to this mapping table to identify the instrument tables to include in the process.

It is recommended that you establish an internal process whereby this procedure is executed after every data load to ensure that mappings are up to date. For more information on the Product to Instrument mapping procedure, see OFS Data Model Utilities User Guide.

d. Select Source from Source Selection.
You can select both Ledger Stat and New Management Ledger Table at the same time. You can select only one Ledger Table per FTP process, either Ledger Stat or the

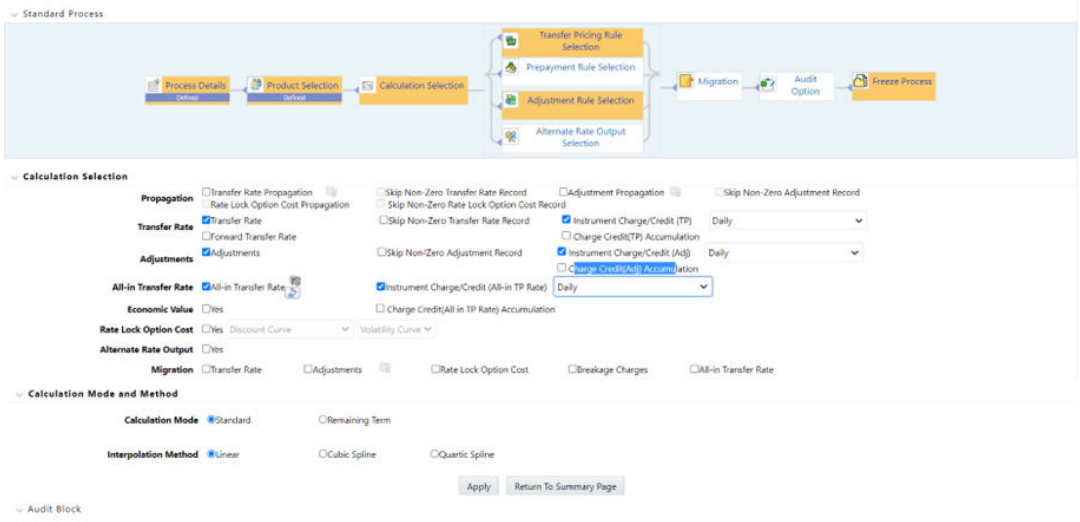
Copyright © 2011, 2026, Oracle and/or its affiliates.

Management Ledger. Multiple Instrument tables are allowed as usual. The engine is using the selected Ledger Table for both Direct Transfer Pricing of Ledger Balances and Ledger Migration. Therefore, the engine will work only on one Ledger Table, not on multiple within any single FTP Process.

e. Select **Apply**.

5. Calculation Selection Block:

Figure 30-4 Calculation Selection



a. Select one or more calculation elements required for the process. You must select at least one calculation item from Transfer Rate, Adjustments, Propagation, Economic Value, or Migration.

b. Choose Skip Non-Zero rates (optional) if you have already populated Transfer Rates or Adjustments through a separate process.

c. Implied forward rates are used for Transfer Pricing forward starting instruments when the "Forward Transfer Rate" calculation selection is checked. Additionally, two conditions in the data must be met for implied forward rates to be used:
Commit Start Date <= As-of-Date

Origination Date > As-of-Date

d. If the Output Forward Rates option is selected in the Audit Block, the Forward Rates are written to an Audit table FSI_IMP_FWD_RATES_AUDIT.

e. Choose the Charge/Credit Calculation (optional), to calculate and output the Instrument level TP charges and credits. If you select the Instrument Charge/Credit Option, you must also choose between Monthly and Daily charge/Credit Accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required (defined within the TP rule for each applicable product/currency) to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.
A checkbox is also available for Charge Credit accumulation only for daily accrual; it will propagate yesterday's accumulated Charge Credit to today's run based on the Holiday Calendar defined in TP/Adjustment rule if any. Today's Charge Credit will be added to yesterday's accumulated Charge Credit and populate the final value to

Copyright © 2011, 2026, Oracle and/or its affiliates.

today's accumulated Charge Credit. If this accumulation flag is not selected, then the accumulated Charge Credit Column would just take today's Charge Credit Value, there would not be any accumulation using yesterday's accumulated Charge Credit.

Charge Credit accumulation is available for all types of (TP, Adjustment, and All_in_TP Rate) Migration of Accumulated Charge Credit to Management Ledger is not supported.

Users can define the Holiday Calendar in TP or the Adjustment Rule for each Product/Ccy combination. For conditional assumption, Holiday Calendar defined at Node Level will only be considered for accumulation; rather than the one provided at Conditional Assumption level.

f. Choose Transfer Rate Propagation/Adjustment Propagation (optional) after selecting Transfer Rate, to pull forward rates from a prior period based on the Propagation Pattern definition.
If you select one or both of the "Instrument Charge/Credit" options for either Transfer Rate or Adjustments, together with Propagation, the process will additionally compute the Account Level Charge/Credit Amounts for the account records, which have propagated rates.

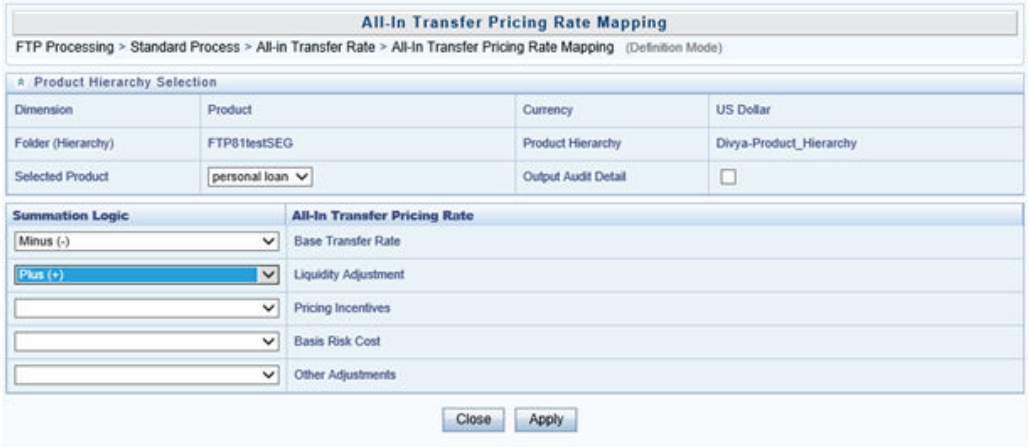
> **Note**
>
> The Propagation process is independent of the Transfer Pricing and Skip non-zero selections. If you select the combination of Propagation + Transfer Pricing and skip non-zero in a single process, the propagation process will execute first pulling forward all rates for eligible instruments. If an instrument already has a TP Rate or Adjustment Rate, the Propagation Process will overwrite the existing value, unless you select the “Skip Non Zero” option on the Propagation line. After Propagation runs, then the remaining instruments will be Transfer Priced and the skip non-zero condition, from the Transfer Rate Line, is applied during this step. The same behavior applies to Adjustment Rate Propagation and calculation.

When Adjustment calculation is selected together with Transfer Rate and TP Rate Propagation, then due to TP Propagation and TP Rate Calculation, the skip non-zero on the TP Rate column is implicitly applied. In this case, during calculation processing, the TP engine skips all records that have a Transfer Rate, even in the context of the Adjustment Rate calculation the Adjustment rate calculations may be missed. That means, when the TP Rate calculation and Adjustment Rate calculation options are selected together, the TP engine will process both assumptions in a single pass (to optimize performance To avoid this issue, users may need to create two separate Transfer Pricing Processes, one each for the Transfer Rate Calculation and another for the Adjustment Rate(s).

g. Choose All-in Transfer Rate (optional) if you want the TP process to calculate and post the All-in Transfer rate for each instrument record. This option allows you to define the aggregation logic for combining any Add-on Rates with the base Transfer Rate. When you select the search icon, you can define for each product the rates to include in the calculation and the related signage for each rate.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-5 All-In Transfer Pricing Rate Mapping - Definition Mode



> **Note**
>
> If you select the Output Audit Trail checkbox, the engine reads this option and generate the output audit data (ALL_IN_TP_RATE) at the time of processing. By default, this checkbox will be unchecked.

h. Choose Economic Value to calculate the Economic Value for each instrument record. By default this optional calculation is unchecked.

- The user can choose to select 'Economic Value' individually or together with other calculation options.
- If the user selects only the 'Transfer Rate' calculation item, then in the 'TP Rule Selection' block, all TP rules, will be displayed.
- If the user selects only 'Economic Value', then in the 'TP Rule Selection' block, only TP rules where Economic Value Calculation inputs are defined (EV_INPUTS_DEFINED flag is on), will be displayed.
- If the user selects both 'Transfer Rate' and 'Economic Value', then in the 'TP Rule Selection' block, only TP rules where Economic Value Calculation inputs are defined (EV_INPUTS_DEFINED flag is on) will be displayed.
- It is also possible to select Economic Value together with Adjustments, Migration, Propagation, Alt Rate Output.

i. Choose Rate Lock Option Cost to calculate the Option Cost for Rate Lock options. Rate Locks are commonly granted to users when they apply for mortgage loans and carry a term of 30, 60, or 90 days. These rate locks fix the interest rate on the loan for the duration of the lock period. Many times, lenders also offer a one-time option for borrowers to take a lower rate if market rates drop during the commitment period. The cost of this option can be calculated and should be charged by the treasury back to the line of business as an internal cost. See Chapter 30 for a full description of this functionality.

j. Choose Alternate Rate Output Mapping (optional), to assign TP results to alternate columns.

k. Choose Migration options (optional), if you want to include migration of your Transfer Pricing results, Breakage Charges, or Rate Lock Option Costs to the Management Ledger table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

l. Choose the Calculation Mode. The default selection is Standard, which applies traditional transfer pricing logic within the process. This entails transfer pricing fixed-rate instruments from the Origination Date (or TP Effective Date if provided) and transfer pricing adjustable-rate instruments from the Last Repricing Date. If the remaining term is selected the effective date for Transfer Pricing all instruments will be the current “As-of-Date”.

> **Note**
>
> If you are calculating Breakage Charges, using the Economic Loss Method, you must select the "Remaining Term" option, to generate the correct cash flows for the funding liability.

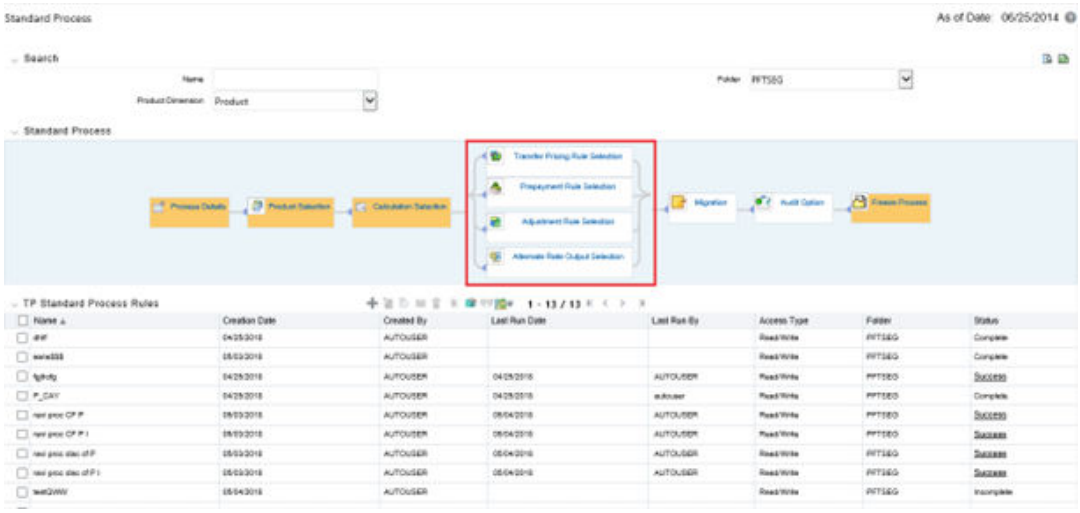
m. Choose the Interpolation Method to specify between Linear (default), Cubic Spline, and Quartic Spline methods.

n. Select **Apply**.

> **Note**
>
> On selecting a calculation element, you will notice the corresponding assumption rule blocks become mandatory (shaded blue).

6. Rule Selection Block:

Figure 30-6 Standard Process - Rule Selection Block



a. Select the assumption rules corresponding to each calculation element. The blocks shaded blue are mandatory and the blocks shaded grey are optional.

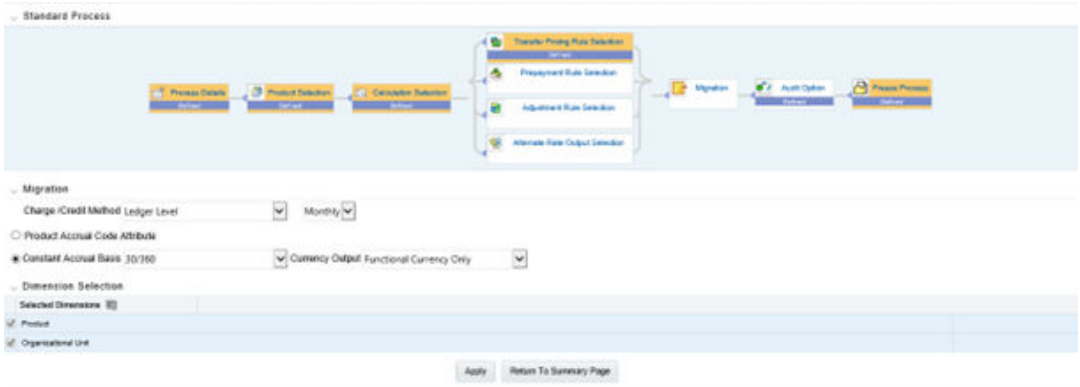
b. After selecting an assumption rule for each process block, select **Apply** to move on.

c. Select **Apply**.

7. Select Migration Parameters:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-7 Standard Process - Migration Parameters



a. Select the Migration Charge / Credit Method. Choose either Account Level or Ledger Level.

b. Choose between Monthly and Daily charge/Credit Accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.

> **Note**
>
> The Accrual Type selection is only available for Account Level migration when the "Daily" accrual option is also selected on the Calculation Selection page.

c. If the Ledger Level Charge / Credit method is selected, then also specify the Accrual Basis source as either the Product dimension attribute or constant value.

d. Select the Currency Output option. Choose either Functional Currency Only or Entered and Functional Currency.

e. Select the Migration Dimensions.
On selecting the Dimension Selection button (see above), Oracle Funds 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.

f. Select **Apply**.

8. Audit Options:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-8 Standard Process - Audit Options



a. Select the Detailed Cash Flows option to output cash flow details for instrument records. This option is used for audit purposes when users want to validate the cash flows being generated by the cash flow engine for certain types of instruments.

b. Select the Number of Records to include in Detail Cash Flows. This input is constrained by the FTP Application Preference setting and controlled by the Administrator. The selection of a large number of instrument records to include in detail cash flow output can result in a very time-consuming process. Within Transfer Pricing, the maximum value for the Number of Records input is 10,000.

c. Select additional parameters for Detailed Cash Flows. The default values for the number of records is “Total Number” and “First n Rows”. It is also possible to specify the Number of Records per Product ID and Random selection as opposed to the First n.

d. Select Implied Forward Rates to Output Relevant Rates to an Audit table. This check box is active only when:

   - The Forward Transfer Rate option has been selected in the Calculation Selection block.
   - The Detailed Cash Flows check box is selected in the Audit block.

   Implied Forward Rates will be output only for relevant forward starting instrument records, which apply to the "Detailed Cash Flows" selection. Therefore, the "Number of Records" input box is used for both "Detailed Cash Flows" and "Implied Forward Rates". It is recommended that users apply a data filter that will highlight specific instrument records that users would like to analyze via Audit output.

   > **Note**
   >
   > Implied Forward Rates are output only for instrument records that have an Origination Date > As-of-Date. In this way, the Number of Records that have implied Forward Rates will always be <= the Number of Records included in Detail Cash Flow output.

e. Select Enable FTP Audit Trail Output to generate the Audit Data at the time of processing. You can select the relevant option for generating audit data for all the data or to read the audit preference from the assumption rule. When Audit Trail output is enabled, details will be written to the FSI_O_FTP_AUDIT_TRAIL table.

f. Select **Apply**.

9. Freeze Process:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-9 Standard Process - Freeze Process



10. Select **Freeze** to complete the process.

> **Note**
>
> Embedded objects like Transfer Pricing, Prepayment or Adjustment Rules can be viewed or edited from the TP process UI itself.

11. Select **Reset** to erase all selections made previously within the Process Definition Flow.
12. Select **Confirm**.

- Transfer pricing for Future Dated Instruments
  Calculating Implied Forward TP rates is possible from an FTP Standard Process.
- Implied Forward Rate Calculation
  An Implied Forward is that rate of interest that is predicted to be the spot rate in the future. For more information, see Rate Locks and the Loan Commitment Pipeline.

30.1.1 Transfer pricing for Future Dated Instruments

Calculating Implied Forward TP rates is possible from an FTP Standard Process.

The base FTP rate quoted on the Commitment Start date is calculated using the Current Forward Rates corresponding to the tenor of the Rate Lock Period requested and the Fixed-Rate Loan period/Payment Frequency based on the TP method selected. If an instrument record has a commit_start_date <= As_of_Date and Origination_Date > As_of_Date, then FTP generates implied forward rates based on the Commit_Start_Date spot IRC Curve.

30.1.2 Implied Forward Rate Calculation

An Implied Forward is that rate of interest that is predicted to be the spot rate in the future. For more information, see Rate Locks and the Loan Commitment Pipeline.

Procedure:

- Identify the record is as a Rate lock record, and select Forward Curve from Calculation Elements (Transfer Rate checkbox as Forward Transfer Rates) under the TP Processing window. If you want to price Transfer rates off the Forward Curve, the Forward Curve can be generated for the same term points as the Spot Curve under the Transfer Price Rule UI. For example, the selected TP Curve has the following term points – 1M, 2M, 3M, 6M, 12M, 2Y, 3Y, then the Forward TP rates will be calculated for the same term points.
- The approach is the same for both Cash Flow Methods and Non-Cash Flow Methods.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- After the Forward TP Curve is ready with all the term points, the TP solution goes through the existing flow for calculating Transfer Rates according to the method selected, except that the Forward TP curve is referenced instead of the Spot TP Curve. If the required tenor or effective date is not directly available, then Interpolation will be used. Three types of Interpolation are available – Linear, Cubic Spline, and Quartic Interpolation.

All TP Methods can reference the appropriate implied forward rates.

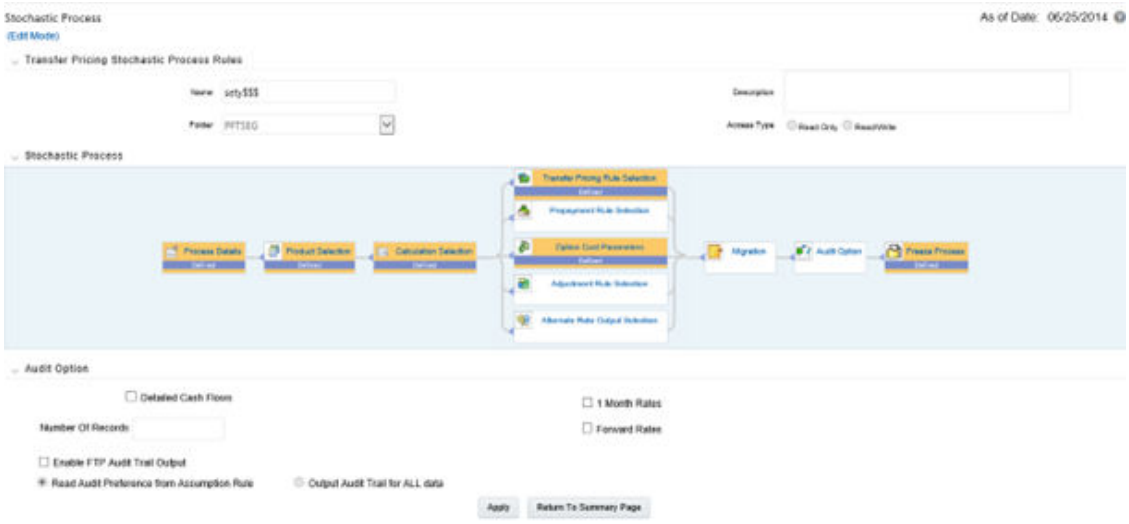
> **Note**
>
> Forward rates are not required for the Spread from Note Rate TP method. Prior day results will be propagated forward, and after post-processing, all open commitments will have a Transfer Rate and Rate Lock Option Cost assigned through either propagation or new calculation. The engine has to use only the Standard Term for TP calculation and not the Remaining Term.

30.2 Creating a Stochastic Transfer Pricing Process

Create a Stochastic Transfer Pricing Process:

- To define and execute Option Cost Processing requests.
- To calculate option cost and Related Charge/Credit Amounts.
- To propagate option costs or adjustment rates for any applicable instrument table from a prior period.
- To migrate charges or credits, for funds provided or used, to the Management Ledger Table.
- To select alternate columns to Output Option Cost results for each record in an instrument table for a Transfer Pricing Process Run.
- To generate Audit Trail Output for Assumption Rule or for All Data.

Figure 30-10 Transfer Pricing - Stochastic Process



Copyright © 2011, 2026, Oracle and/or its affiliates.

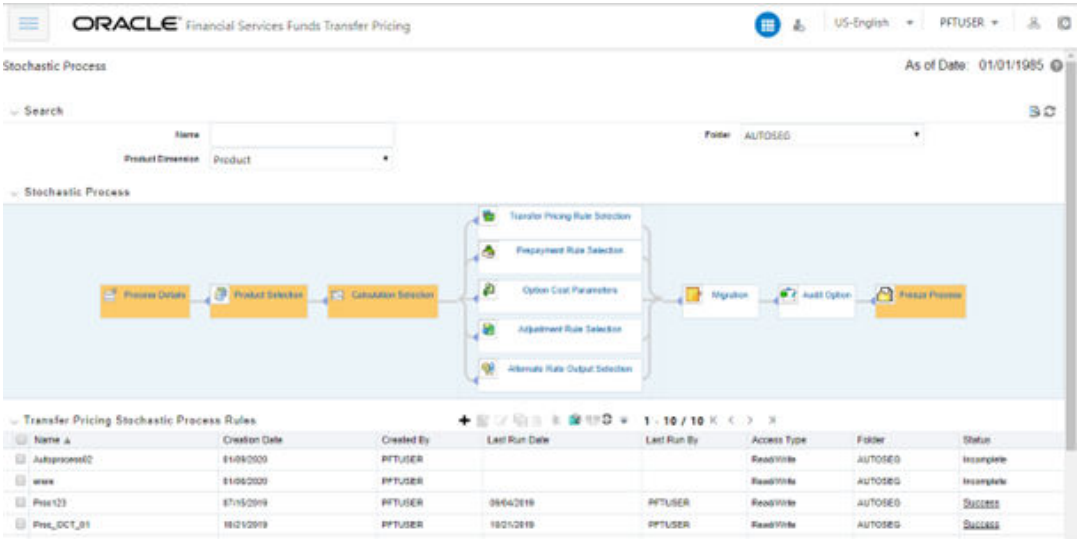
The prerequisites for defining and executing the stochastic transfer pricing processing requests are:

- Performing basic steps for creating or editing a Transfer Pricing rule.
- Performing basic steps for creating or editing an Adjustment rule.
- Performing basic steps for creating or editing a Stochastic Rate Indexing Rule.

To create and execute the stochastic transfer pricing process:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Processing**, and then select the **Stochastic Process**.

Figure 30-11 FTP Processing - Stochastic Mode summary page



The following table describes the key terms used for this procedure:

Table 30-2 Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Product Hierarchy	Displays the selected product hierarchy. The Product Hierarchy allows you to select Product Members to include in the Process. Note, you have two methods for selecting data that will be included in the process. You can select Parents / Children from a hierarchy or you can use Source selection.
Source	Allows you to select one or more source tables to include in your process. If sources are selected directly then the Product Hierarchy will be disabled and selections made within the hierarchy will be lost. Users can utilize one method or the other to define data that will be included in the process, but not both methods at the same time.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Filter	Filters allow you to restrict your data selection based on any attribute that exists within an instrument table. You define filters under Common Object Maintenance and reference your filter within the Product Selection block of your Process. The supported Filter Types are Attribute Filter, Data Filter, Hierarchy Filter, and Group Filter.
Transfer Pricing rule	This LOV allows you to select a Transfer Pricing rule. The Transfer Pricing rule provides the application with assumptions that are applied to products to calculate their transfer rates and option costs.
Prepayment Rule	This LOV allows you to select a Prepayment rule. The Prepayment rule provides the application with prepayment assumptions that you want to apply to products in conjunction with cash flow transfer pricing methods.
Term Structure Model	An Option Cost parameter, 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 is used to interpolate rates on the valuation curve for terms that fall between given points.
Stochastic Rate Index Rule	An Option Cost parameter 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 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Random Number Generation Method	An Option Cost parameter determines how the Monte Carlo process selects random numbers. The Random Number Generation Method has two variations: • **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. • **Pseudorandom Sequences**: These are the traditional random numbers generated by most compilers. They are designed to do well on some statistical tests: low autocorrelation, the high period before the sequence repeats itself.
Adjustments Rule	This LOV allows you to select an Adjustments rule. The Adjustments rule lets you apply transfer pricing add-on rates to an instrument record for purposes of determining specific charge and credit amounts.
Alternate Rate Output Mapping Rule	This LOV allows you to select an Alternate Rate Output Mapping rule. The Alternate Rate Output Mapping rule lets you select the alternate columns to output option cost, and adjustment calculation results for each record in an instrument table for a stochastic transfer pricing process run. This functionality allows you to output more than one option cost, or adjustment calculation result for each record in the instrument table through multiple transfer pricing process runs.
Calculation Method	Within the Calculation Selection block, there are two calculation options: Standard or Remaining Term. Standard is the most commonly used method and applies traditional transfer pricing logic. The remaining Term allows you to assign Transfer Rates based on current period rates.
Smoothing Method	A calculation selection parameter, Smoothing Method allows users to decide between Linear, Cubic Spline, or Quartic Spline interpolation methods. This selection affects how rate lookups happen for terms that fall between anchor points on your Interest Rate Curves.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Charge Credit Method	A migration block parameter, the Charge Credit Method selection determines how the Ledger Level TP Charge or Credit is calculated during the migration process. There are two options available: • **Instrument Level**: With this method, the instrument level charges and credits that were posted during the Stochastic process are added and posted to the Ledger table, together with the weighted average Option Adjusted Spread and Static Spread. • **Ledger Level**: with this method, the balance of the instrument (average, ending, or other) to arrive at weighted average rates weights the instrument level OAS and Static Spread. The weighted average rates are migrated to the Ledger table and multiplied by the corresponding Ledger Balance (average or ending) to arrive at the charge/credit amount.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Charge/Credit Accrual Basis	Select the Charge/Credit Accrual Basis on the Transfer Pricing Process Migration block or, define the Accrual Basis as an attribute for each Product dimension member in the Dimension Management – Members screen. In case no selection is made, a default Accrual Factor of Actual/Actual should be applied. Charge/Credit Accrual Basis denotes the basis on which the Ledger level interest accrual is calculated. You need to select the accrual factor to be applied when calculating the Ledger Level cost of funds. For instrument-level charges and credits, the accrual basis from the instrument record is always used. When the Ledger Level charge/credit method is selected, the cost of funds is calculated using the following formula: *Option Cost Charge = (Static spread - OAS) x Balance x (1 - percent sold/100) x accrual basis* The type of balance is determined by the TP Charge Credit Balance selection made within TP Application Preferences. Choices include Average Balance, Ending Balance, or Other Balance. Oracle Funds Transfer Pricing offers the following accrual basis options: • **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 the number of days in the month divided by 360 days. • **Actual/Actual**: Applies the accrual basis calculation of the number of days in the month divided by the number of days in the year. • **30/365**: Applies the accrual basis calculation of 30 days divided by 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 the number of days in the month divided by 365 days. • **Business/252**: Applies the accrual basis calculation of the number of business days in the month divided by 252 days. A Holiday calendar selection is required if the business/252 accrual basis is selected. If the holiday calendar is not selected, the engine considers Accrual type ACT/ACT as a default for calculation.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Accrual Type	A Calculation Selection Parameter and Migration Block Parameter. Users have the option to choose between Monthly and Daily charge/credit accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.
Currency Output	A migration block parameter allows you to select the output currency. Oracle Funds Transfer Pricing offers you the following currency output options: • Entered and Functional Currency • Functional Currency Only 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 the US dollar to display its consolidated annual results. In this case, the US dollar is the functional currency. In other words, the currency in which an organization keeps its books is its functional currency.
Migration Dimensions	Dimensions for which you want to migrate charges or credits, for funds provided or used, to the Management Ledger Table. Oracle Funds Transfer Pricing provides you with a Shuttle Control component to specify migration dimensions.
Available Dimensions	One of the two Shuttle Control windows, it contains the names of the dimensions available, in addition to the selected Product, for inclusion during the migration process. The dimensions available are: • Common Chart of Account • Organizational Unit • GL Account • Product • Legal Entity **Note** Only the Management Table processing key dimensions are available for inclusion during the migration process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

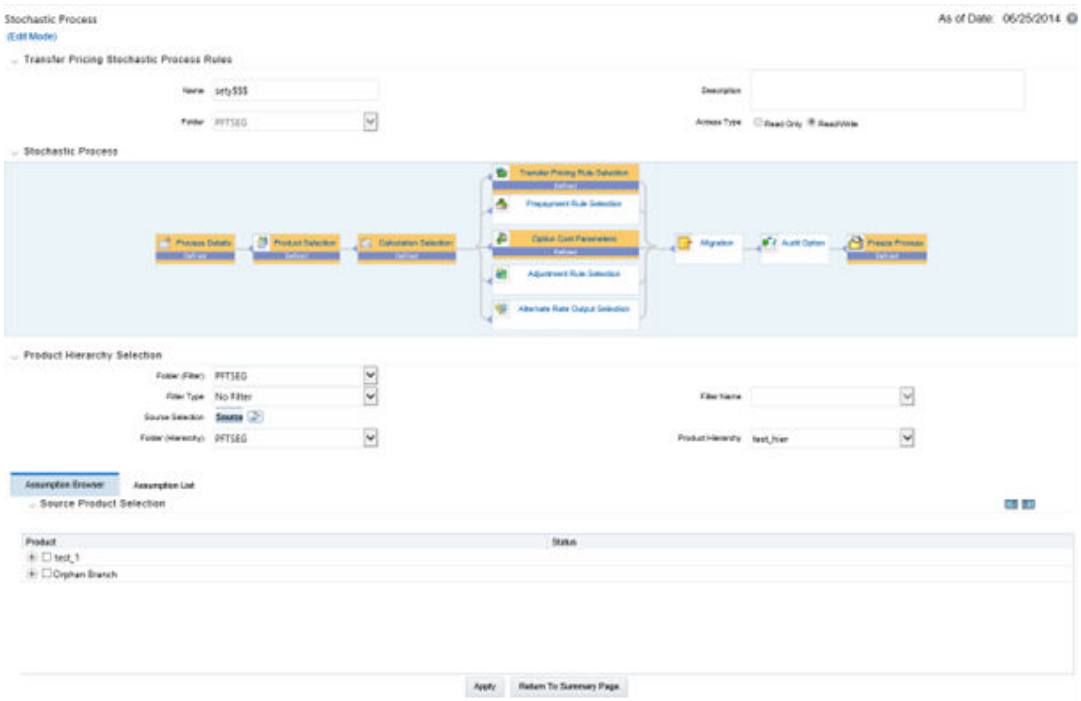
Table 30-2 (Cont.) Fields and Descriptions from the FTP Processing - Stochastic Mode summary page

Term	Description
Selected Dimensions	One of the two Shuttle Control windows, it contains the names of the dimensions, in addition to the selected Product dimension that has already been selected for migration.
Audit Options	Within the Audit Options block, you have the option to output detailed cash flows. Select the check box to enable this option and define the number of instrument records for which you would like to output details. Note, the Administrator can limit the maximum number of instruments for which you can compute cash flows because this can be a time-consuming process. In any case, the maximum allowable value is 10,000 instruments. Additionally, the Enable FTP Audit Trail Output option allows you to select the relevant option to either reading the Audit Preference from the assumption table or generating an Audit Trail for all data. If you select the check box Enable FTP Audit Trail Output, then it is mandatory to select one of the options for the Audit Trail. When you select the Enable Audit Trail Output option, by default Read Audit Preferences from Assumption Rule is selected.
Freeze Process	The freeze process block allows you to finalize the assumptions made in the Process definition flow or to clear all assumptions.

2. Complete standard steps for this procedure.
3. Additional Steps to Specify Stochastic Transfer Pricing Process Parameters.
4. Product Selection Block:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-12 Stochastic Process - Product Selection



a. Select a Product Hierarchy and select nodes from the hierarchy corresponding to data you want to include in the process.

b. Alternatively, select the Source Selection button, and select the instrument tables corresponding to the data you want to include in the process.

c. Select a Filter (optional), to constrain the data to be included in the process. Filters work as a secondary constraint, applied after the data set is determined based on Product Hierarchy Member Selection or Source Selection.

> **Tip**
>
> Before using the product hierarchy approach for selecting data to include in your process, there is a procedure that must be Run ("PRODUCT TO INSTRUMENT MAPPING"). This procedure can be executed from the Batch Scheduler – Run - interface.

The purpose of the Product to Instrument mapping procedure is to scan all instrument tables (FSI_D_xxx) and populate the mapping table ("FSI_M_PROD_INST_TABLE_MAP") with a listing of the product dimension members that exist within each instrument table. When you select Products (parents or children) within an ALM or FTP process definition, the process refers to this mapping table to identify the instrument tables to include in the process.

It is recommended that you establish an internal process whereby this procedure is executed after every data load to ensure that mappings are up to date.

For more information on the Product to Instrument Mapping Procedure, see OFS Data Model Utilities User Guide.

d. Select Source from Source Selection.

Copyright © 2011, 2026, Oracle and/or its affiliates.

You can select both Ledger Stat and New Management Ledger table at the same time. You can select only one Ledger Table per FTP process, either Ledger Stat or the Management Ledger. Multiple Instrument tables are allowed as usual. The engine is using the selected Ledger Table for both Direct Transfer Pricing of Ledger Balances and Ledger Migration. Therefore, the engine will work only on one Ledger Table, not on multiple within any single FTP Process.

e. Select **Apply**.

5. Calculation Selection Block:

   a. Select one or more calculation elements required for the process. You must select at least one calculation item from Option Cost, Adjustments, or Migration.

   b. Choose Skip Non-Zero rates (optional) if you have already populated Option Costs or Adjustments through a separate process.

   c. Choose the Charge / Credit calculation (optional), to calculate and output the Instrument level TP charges and credits. If you select the Instrument Charge/Credit option, you must also choose between Monthly and Daily charge/credit accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required (defined within the TP Rule for each applicable product/currency) to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.

   d. Choose Option Cost Propagation/Adjustment Propagation (optional) after selecting Option Cost, to pull forward rates from a prior period based on the Propagation Pattern definition.

   e. Choose All-in Transfer Rate (optional) if you want the TP process to calculate and post the All-in Transfer Rate for each instrument record. This option allows you to define the aggregation logic for combining any Add-on Rates with the base Transfer Rate. When you select the search icon, you can define for each product the rates to include in the calculation and the related signage for each rate.

Figure 30-13 Stochastic Process - All-In Transfer Pricing Rate Mapping Definition Mode



Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> If you select the Output Audit Trail checkbox, the engine reads this option and generate the output audit data (ALL_IN_TP_RATE) at the time of processing. By default, this checkbox will be unchecked.

f. Choose Alternate Rate Output Mapping (optional), to assign Option Cost results to alternate columns.

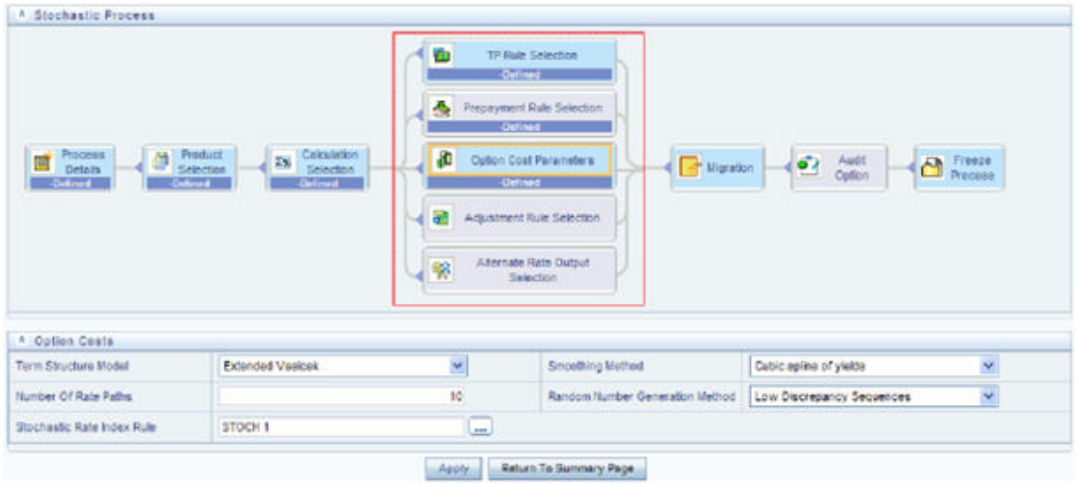
g. Choose Migration Options (optional), if you want to include migration of your Option Cost results to the Management Ledger Table.

h. Choose the Calculation Mode. The default selection is Standard, which applies traditional transfer pricing logic within the process. This entails calculating transfer rates for Fixed-Rate Instruments from the Origination Date (or TP Effective Date if provided) and Transfer Pricing Adjustable-Rate Instruments from the Last Repricing Date. If the remaining term is selected, the effective date for Transfer Pricing all instruments will be the current “As-of-Date”.

i. Select **Apply**.
On selecting a calculation element, you will notice the corresponding assumption rule blocks become mandatory (shaded blue).

6. Rule Selection Blocks:

Figure 30-14 Stochastic Process - Rule Selection



a. Select the assumption rules corresponding to each calculation element. The blocks shaded blue are mandatory and the blocks shaded grey are optional.
The standard assumption rules are included in this step. Notice the Stochastic Process Flow has one additional Parameter Block, Option Cost Parameters:

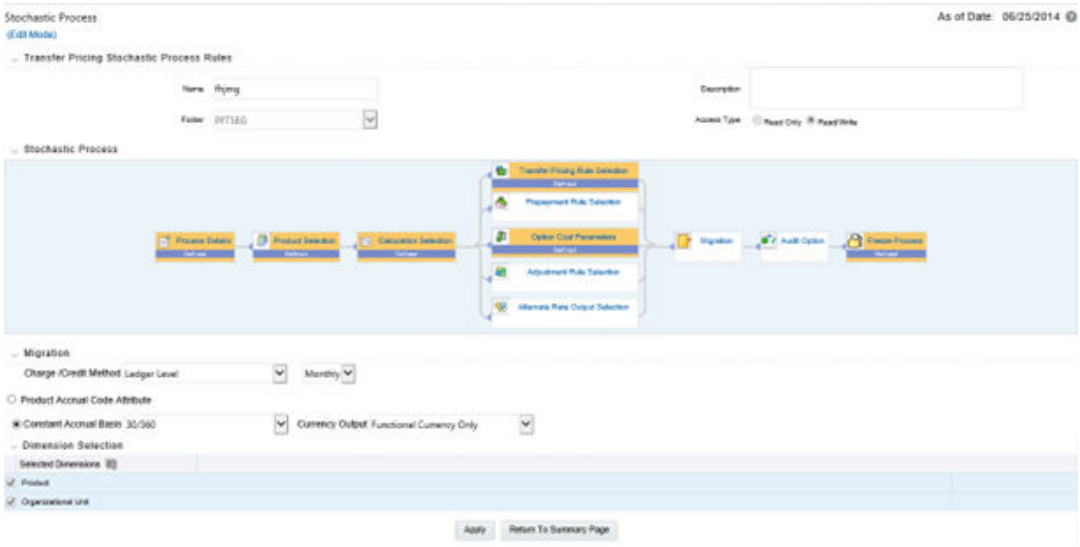
- Select a Term Structure Model
- Select the Number of Rate Paths
- Select the Smoothing Method
- Select the Random Number Generation Method
- Select a Stochastic Rate Indexing Rule

Copyright © 2011, 2026, Oracle and/or its affiliates.

b. Select **Apply**.

7. Select Migration Parameters:

Figure 30-15 Stochastic Process - Migration Parameters



a. Select the Migration Charge/Credit Method. Choose either Account Level or Ledger Level.

b. Choose between Monthly and Daily charge/Credit Accrual. The default selection is Monthly. If Daily is selected, a holiday calendar is also required to determine the number of days to accrue. For example, on weekdays (Monday - Thursday), 1 day of accrual will be posted and on weekends (Friday), 3 days of accrual will be posted at Friday's rate. When additional holidays are encountered on any other day of the week, similar logic is applied to extend the number of accrual days accordingly.

> **Note**
>
> The Accrual Type selection is only available for Account Level migration when the "Daily" accrual option is also selected on the Calculation Selection page.

c. If the Ledger Level Charge / Credit method is selected, then also specify the Accrual Basis source as either the Product dimension attribute or constant value.

d. Select the Currency Output option. Choose either Functional Currency Only or Entered and Functional Currency.

e. Select the Migration Dimensions.

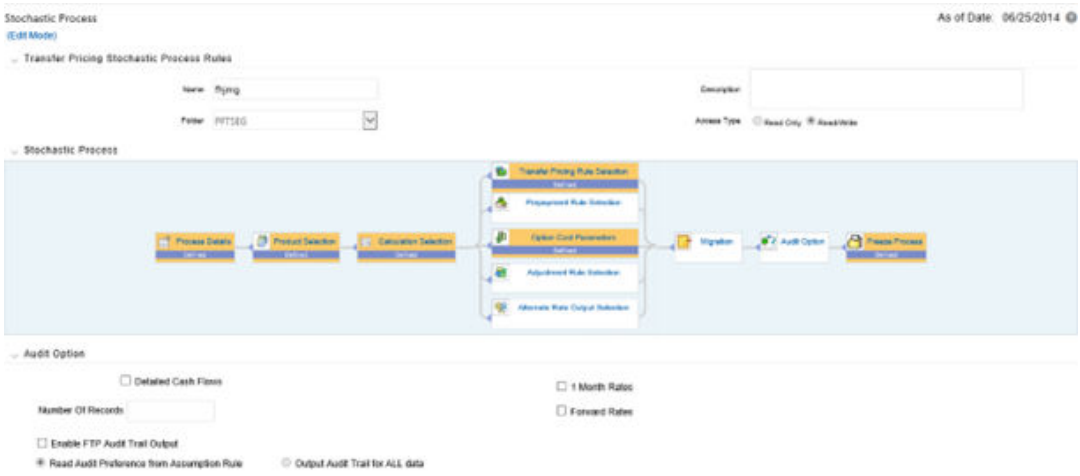
Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> On selecting the Dimension Selection button (see above), Oracle Funds 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.

f. Select **Apply**.

8. Select Audit Options:

Figure 30-16 Stochastic Process - Audit Options



a. Select the Detailed Cash Flows option to output cash flow details for instrument records. This option is used for audit purposes when users want to validate the cash flows being generated by the cash flow engine for certain types of instruments.

b. Select the Number of Records to include in detail cash flows. This input is constrained by the FTP Application Preference setting and controlled by the Administrator. The selection of a large number of instrument records to include in detail cash flow output can result in a very time-consuming process. Within Transfer Pricing, the maximum value for the Number of Records input is 10,000.

c. Select additional parameters for detailed cash flows. The default values for the number of records is “Total Number” and “First n Rows”. It is also possible to specify the number of records per Product ID and Random selection as opposed to the First n.

d. Select 1 Month Rates and Forward Rates options to output rate details which can be accessed by the Business Intelligence applications for reporting.

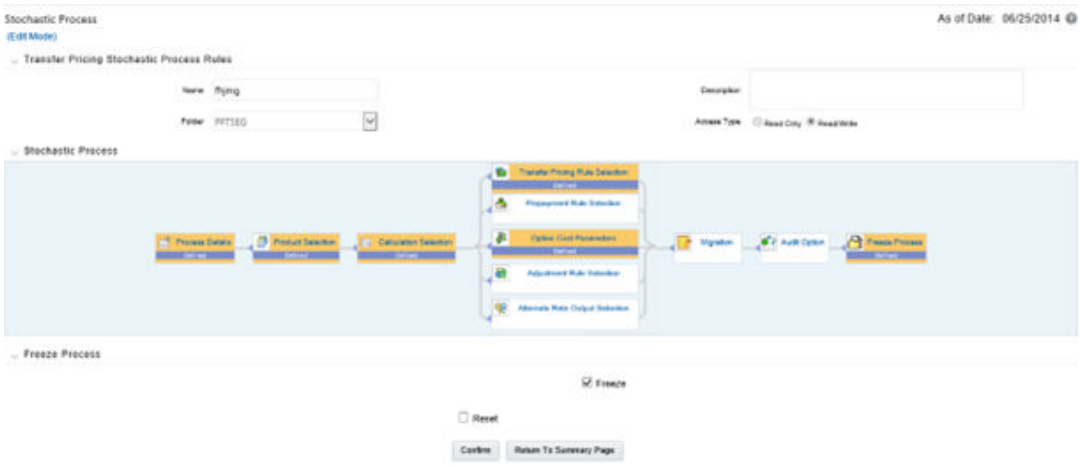
e. Select Enable FTP Audit Trail Output to generate the audit data at the time of processing. You can select the relevant option for generating audit data for the entire data or read the audit preference from the assumption rule.

f. Select **Apply**.

9. Freeze Process:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 30-17 Stochastic Process - Freeze Process



a. Select **Freeze** to save the process.

b. Select **Reset** to erase all selections made previously within the Process Definition Flow.

10. Select **Confirm**.

30.3 Execute a Transfer Pricing Process

You execute a Transfer Pricing Process:

- To generate Transfer Rates, TP Add-on Rates, or Option Cost Results.
- To propagate Transfer Pricing Results for any applicable instrument table from a prior period.
- To migrate charges or credits, for funds provided or used, to the Management Ledger table.
- To output, in pre-selected alternate columns, Transfer Rate, Option Cost, and Adjustment Calculation results for each instrument record in an account table for a Transfer Pricing Process run.

Executing a Transfer Pricing Process involves specifying the parameters necessary for successfully running an active version of the rule, and running the version.

The prerequisites for executing a Transfer Pricing process are performing basic steps for creating or editing a Standard Transfer Pricing process or a Stochastic Transfer Pricing process.

To execute a transfer pricing process:

1. From the LHS menu, select **Funds Transfer Pricing**, select **FTP Processing**, and then select the relevant **Transfer Pricing Process**.
2. Select a Process that you want to execute.
   The status column indicates whether a process can be run. The following are the possible status conditions:
   - **Complete**: Indicates the process is fully defined and ready to be run.
   - **Incomplete**: Indicates the process is partially defined and cannot be run.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **View Log**: Indicates the process has already been run and also indicates a process can be re-run. In this case, View Log is a hyperlink that allows you to navigate to the Log Viewer page where you can review any processing errors or alerts related to this process.

Figure 30-18 Transfer Pricing Stochastic Process Rules



3. After executing the process (above), select the View Log Hyperlink.

Figure 30-19 View Log



4. Select the Task ID (also known as the Unique System Identifier) to view a report for any processing errors.

> **Note**
>
> If significant processing errors exist, you should re-run your process.

The Transfer Pricing process is complete. You can access instrument-level results and Management Ledger results through either the Enterprise Financial Performance Analytics BI application or the Asset Liability Management Analytics BI application.

- Undo
  The Undo functionality allows you to undo the results of a recently executed FTP process. This is useful if you have accidentally executed a TP process rule.

30.3.1 Undo

The Undo functionality allows you to undo the results of a recently executed FTP process. This is useful if you have accidentally executed a TP process rule.

This component will work with the new Management Ledger table structure. With the new structure, Management Ledger will work correctly to support multiple UNDO's given that each incremental process will insert rather than update. Because of this, you can UNDO any process.

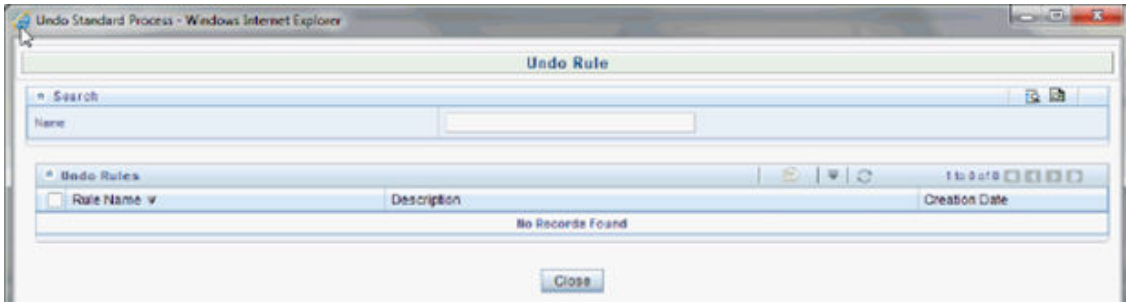
The UNDO feature is applicable to both Standard and Stochastic Processes. To execute the UNDO process, click the **Undo** icon on the TP Process summary page.

The Undo window is displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Search for the TP Process rule that you want to undo. The rule is displayed in the Undo Rules list. Select a rule and click **Undo**.

Figure 30-20 Undo Rules



Copyright © 2011, 2026, Oracle and/or its affiliates.

31
Break Identification Process

Breaks are associated with Assets and Liabilities that have fixed maturities and have experienced a full prepayment or pre-closure, partial prepayment, or restructuring. Any event that causes the bank to receive a change to scheduled contractual cash flows on a fixed maturity instrument results in a Break Funding Event and should be evaluated. Transactions that could cause a change in future cash flows would include full loan prepayments, partial loan prepayments, early withdrawal of term deposits, or a change in maturity tenor, payment amount, payment frequency, or other contractual terms.

The Break Identification Process allows you to perform the following tasks:

- Determine the data that you want to process (Product Selection block).
- Specify the parameters for the process. The parameters include break types like a full break, partial break, and change in attributes.
- Execute or Run the Break Identification Request and generate results (Break Identification Process summary page).
- Create a Break Identification Process
  To define and execute a Break Identification Process. The Break Identification Process will compare current period-instrument data with prior period-instrument data to identify break events. When breaks are detected, the related instrument records are copied into the FSI_D_BREAK_FUNDING_CHARGES table. This table then becomes the source table for further FTP Adjustment Rule > Breakage Charge processing.
- Executing a Break Identification Process
  You execute a Break Identification to compare the current period and prior period data to identify different break types.
- Performance Tuning Break Identification Processes
  Processing time for Break Identification Processes can be reduced by modifying parallel processing parameters in the FSI_PARALLEL_PARAMETERS table.

31.1 Create a Break Identification Process

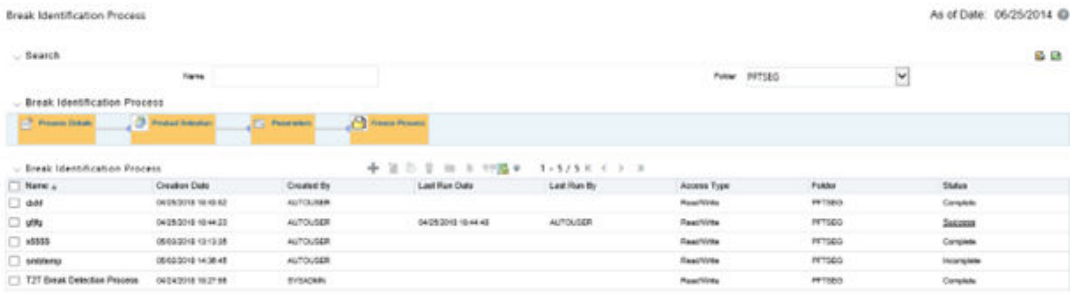
To define and execute a Break Identification Process. The Break Identification Process will compare current period-instrument data with prior period-instrument data to identify break events. When breaks are detected, the related instrument records are copied into the FSI_D_BREAK_FUNDING_CHARGES table. This table then becomes the source table for further FTP Adjustment Rule > Breakage Charge processing.

To create a Break Identification Process:

1. Navigate to the Standard Break Identification Process summary page and create a new Break Identification Process.

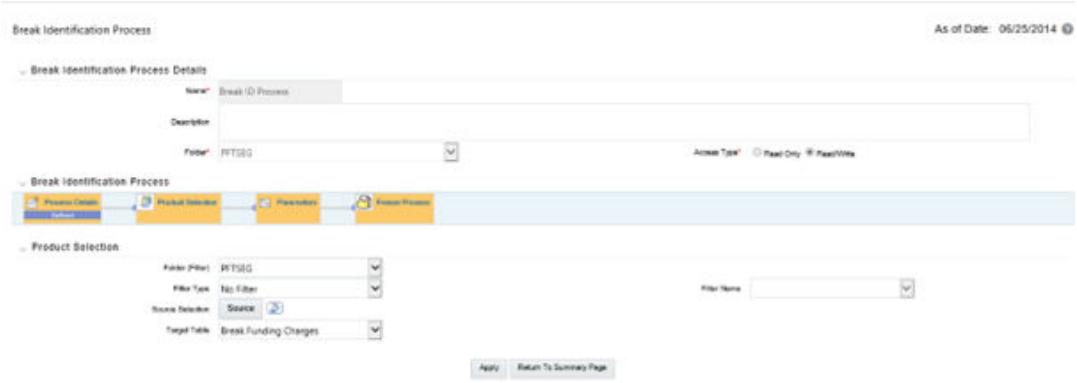
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 31-1 Break Identification Process summary page



2. Select the Product Selection block.

Figure 31-2 Break Identification Process Details



3. Enter or select the following:

Table 31-1 Fields and Descriptions from the Break Identification Process Details page

Term	Description
Folder	The folder where you can save the definition. You can give other users, read/write, or read-only privileges.
Filter	Filters allow you to restrict your data selection based on any attribute that exists within an instrument table. You define filters under Common Object Maintenance and reference your filter within the Product Selection block of your Process. The choice of the data filter would determine the instrument records that should be picked up from the As-of-Date and the prior period date for comparative analysis. The supported Filter Types are: • Attribute Filter • Data Filter • Hierarchy Filter • Group Filter **Note**: Data Filters with Expressions are not supported for Break Identification.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 31-1 (Cont.) Fields and Descriptions from the Break Identification Process Details page

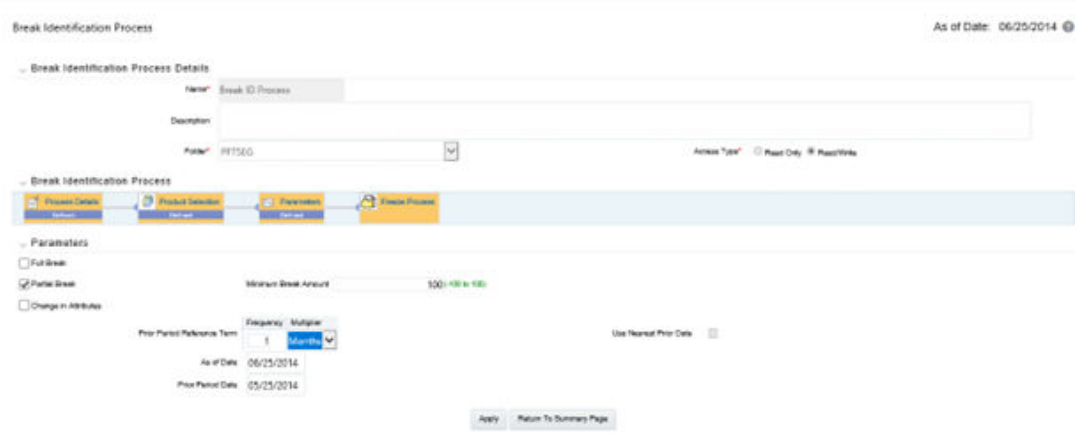
Term	Description
Source	Allows you to select one or more source Instrument tables to include in your process. Based on the Instrument Table(s) selected, the instrument records on the As-of-Date and the prior period date are chosen for comparison.
Target Table	Indicates the destination table where break event records will be posted. The default (seeded) table is the FSI_D_BREAK_FUNDING_CHARGES table. Users can additionally register user-defined tables for posting Breakage Funding records if needed.
Parameters	There are three types of break parameters for the accounts: • **Full Break**: Fully repaid or terminated accounts are considered as a Full Break. • **Partial Break**: Partly repaid accounts are considered as Partial Break. • **Change in Attributes**: Here a restructure of the Instrument record happens due to a change in critical attributes or terms other than Balance. You can execute these breaks individually or together.
Minimum Break Amount	Minimum Break applies to both Positive and Negative breakage amounts. If the user enters the minimum break as 1000, it means that the minimum break amount ranges from -1000 to +1000. If the Breakage Amount that is calculated is less than or equal to the Minimum Break Amount, then it is not passed to the Break Funding Charges table.
Freeze Process	The freeze process block allows you to finalize the assumptions made in the Process definition flow or to clear all assumptions.

- Select a Filter (optional) to constrain the data to be included in the process. The supported Filter Types are Attribute Filter, Data Filter, Hierarchy Filter, and Group Filter.
- Select the source table(s) that you want to include in the process.
- Select the target table, which is a Break Funding Instrument table.

4. Select the Parameters block.

Copyright © 2011, 2026, Oracle and/or its affiliates.

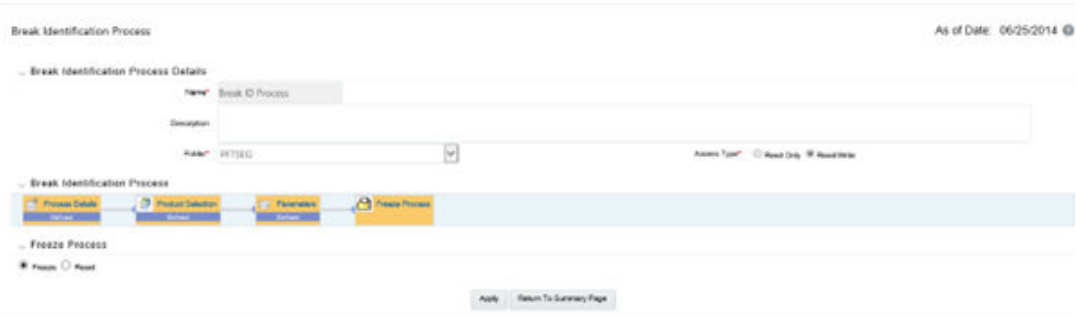
Figure 31-3 Break Identification Process Details page



- Select the type of break that you would like to search for and fill in the related details. In the case of a Full Break, the filter will work only on the prior period-instrument record. In the case of a Partial Break or Change in Attributes, the filter will work on both the prior period and current period-instrument records.
- Input the **Minimum Break Amount** as a positive value. The engine will apply the absolute value of the amount of input ranging from - input amount to + input amount. For example, if the input is 100, then break amounts between -100 and +100 will be excluded. This input allows you to filter very small/insignificant break amounts, reducing the amount of data copied into the Break Funding Charges table.
  Note there are two approaches for determining the Prior Period Date. You can input the Prior Period Reference Term and based on the current As of Date, the Prior Period Date will be calculated, or you can select the “Use Nearest Prior Date” option, and the engine will then look back at the historical data to determine the nearest prior As of Date and will use this as Prior Period Date.

5. Select the **Freeze Process** block.

Figure 31-4 Freeze Process block



- Select **Freeze** to complete the process.
- Select **Reset** to erase all selections made previously within the process definition flow.
- Select **Confirm**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

31.2 Executing a Break Identification Process

You execute a Break Identification to compare the current period and prior period data to identify different break types.

To execute a Break Identification Process:

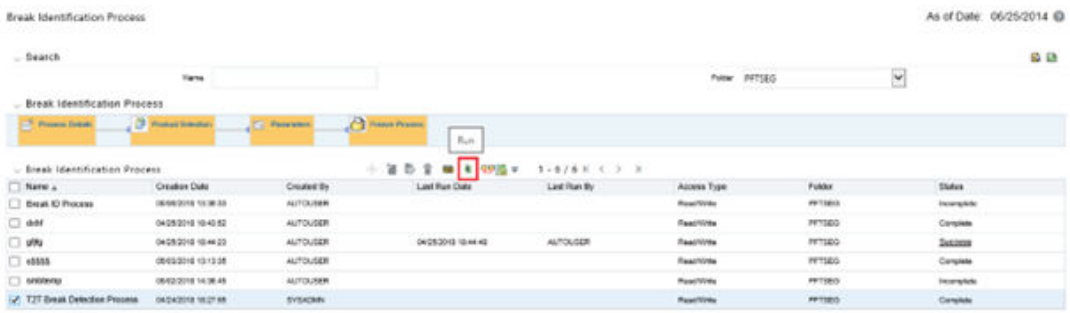
1. Perform the basic steps for Creating a Break Identification Process.
2. There are two approaches to execute the Break Identification Process:ol
   - UI Based Procedure
   - Batch Based Procedure

UI Based Procedure

To execute the Break Identification Process using UI:

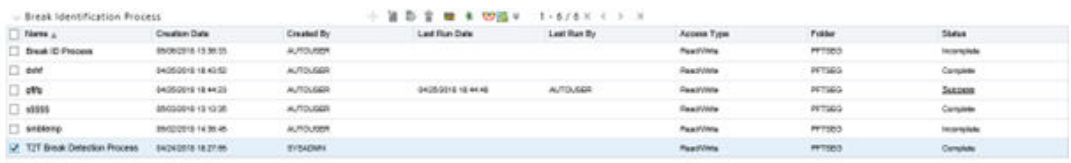
1. Navigate to the Break Identification Process summary page.

Figure 31-5 Break Identification Process summary page



2. Select a Process that you want to execute or Run.
   The status column indicates whether a process can be Run. There are three possible status conditions:
   - **Complete**: Indicates the process is fully defined and ready to be Run.
   - **Incomplete**: Indicates the process is partially defined and cannot be Run.
   - **View Log**: Indicates the process has already been Run and also indicates a process can be re-Run. In this case, View Log is a hyperlink that allows you to navigate to the Log Viewer page where you can review any processing errors or alerts related to this process.

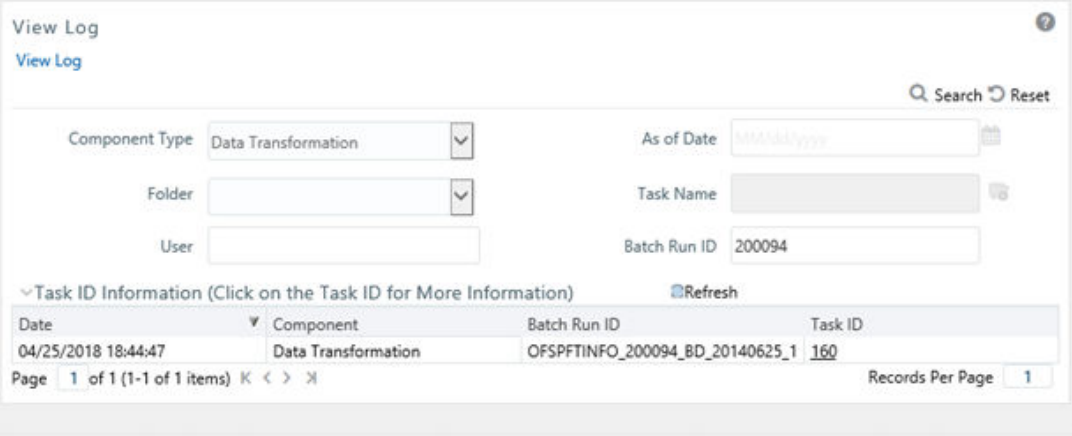
Figure 31-6 Break Identification Processes View



3. After executing the preceding process, select the View Log hyperlink.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 31-7 View Log



4. Select the Task ID (also known as the Unique System Identifier) to view a report for any processing errors.

> **Note**
>
> If significant processing errors exist, you should re-Run your process.

The Break Identification process is complete.

Batch Based Procedure

Break Identification Processes can also be executed as batch processes. The following approaches are available:

- Simplified Batch
- ICC Batch

Using Simplified Batch

To execute the Break Identification Process using simplified batch:

1. Navigate to **Financial Services Applications**, select **Administration**, and then select **Simplified Batch**.
2. Create a **New Simplified Batch**.
3. Select Task Type **Transform Data**.
4. Select **Break_Detection** from the list of seeded process types.
5. Input the following in the Optional Parameters text box:
'Break Detection Process System ID','User ID', for example, '200883','TPUSER'
6. Save the rule and execute.

Using ICC Batch

To execute the Break Identification Process using ICC batch:

1. Navigate to **Operations** and select **Batch Maintenance**.
2. Create a new batch.

Copyright © 2011, 2026, Oracle and/or its affiliates.

3. Select the **Batch Name** to add the task.
4. Click the **Add** button under the **Task Details** section.
5. Define the **Task ID** and **Description**.
6. Select **Components** as **Transform Data**.
7. Select **Break_Detection** from the list of seeded process types.
8. Input the following required parameters:
   Break Detection Process System ID and User Name. These parameters should be given in the following format (sys ID, User ID), for example, 101000, TPUSER.
9. Save the rule and execute.

- Run Rule Framework
  You can use the Run Rule Framework to execute the Break Identification Process.

31.2.1 Run Rule Framework

You can use the Run Rule Framework to execute the Break Identification Process.

To execute the rule using the Run Rule Framework

1. Navigate to **Rules Framework** and then select **Process**.
2. Create a new rule.
3. Define the **Folder**, **Code**, and **Name of rule**.
4. Select the **Component** as **Transform Data**.
5. Select **Break_Detection** from the list of seeded process types.
6. Input the following required parameters:

   Break Detection Process System ID and User Name. These parameters should be given in the following format (sys ID, User ID), for example, 101000, TPUSER.
7. Save the rule and execute.

31.3 Performance Tuning Break Identification Processes

Processing time for Break Identification Processes can be reduced by modifying parallel processing parameters in the FSI_PARALLEL_PARAMETERS table.

Parallel execution-related parameters for Break Identification are seeded by the application with a process_engine_cd value of "3" in the FSI_PARALLEL_PARAMETER table.

The Break Identification process looks for the value given in the parallel query field. If the value is given as "ENABLED", the engine uses the PARALLEL_DEGREE input and sets the degree of parallelism for the execution of the queries within the Break Identification Engine.

The user or implementation team should set these values when they want to take advantage of parallel query executions. The optimal value for the PARALLEL_QUERY input depends on the environment and should be iteratively tested to arrive at the appropriate value.

Copyright © 2011, 2026, Oracle and/or its affiliates.

32

Rate Locks and the Loan Commitments Pipeline

A Rate Lock is a lender's promise to hold a certain interest rate for the borrower, usually for a specified period of time and fee, while the loan application is being processed.

Rate locks are commonly granted to borrowers when they apply for a mortgage loan and carry a term of 30, 60, or 90 days. In Oracle Funds Transfer Pricing, these loan commitments (which are not yet on the balance sheet) are stored in the FSI_D_LOAN_COMMITMENTS table, separate from loans that are already funded. These loan commitments can be Transfer Priced using implied forward rates, which correspond to the assumed loan start date (end of commitment period). This capability allows the treasury to “lock-in” a loan-funding rate at a point in time before the actual loan funding. For more information on using the Forward Rates option, see the Transfer Pricing Process.

- Rate Lock Options
  Many times, lenders also offer a one-time option for borrowers to take a lower rate if market rates drop during the commitment period.
- Transfer Price using Forward Rates and Calculate Rate Lock Option Costs
  The Standard FTP Process provides setup options that allow you to Transfer Price Data in the Loan Commitments table using Forward Rates and calculate the Related Rate Lock option costs. To do this, please select the Loan Commitment table under Source Selection.
- Rate Management
  As shown in the TP Process, the Rate Lock Option Cost calculation requires two inputs both of which come from the Rate Management > Interest Rates page.
- Implied Forward Rates Output
  Users have the option to output the implied forward rates to an Audit Table if needed for validating the Transfer Rate result. To support the validation requirement, the FTP engine (optionally) writes the forward rates to an Audit table - FSI_O_FTP_IMP_FWD_RATES_AUDIT. The option to write implied Forward Rates is available in the Standard TP Process on the Audit block.
- Transfer Pricing Future Dated Instruments (Loan Commitments)
  The base FTP rate quoted on the Commitment Start date is calculated using the current Forward Rates corresponding to the tenor of the Rate Lock period requested and the loan period/payment frequency based on the TP method selected. If an instrument record has a commit_start_date <= as_of_date and origination_date > as_of_date, then FTP will generate implied forward rates based on the commit_start_date spot IRC curve.
- Execution and Results
  You can execute the TP Process to calculate Forward Rates and Rate Lock Option Costs by selecting the process in the summary screen and then click **Run**.

32.1 Rate Lock Options

Many times, lenders also offer a one-time option for borrowers to take a lower rate if market rates drop during the commitment period.

Copyright © 2011, 2026, Oracle and/or its affiliates.

If on the Settlement Date, the advertised rate for the chosen fixed-rate period falls below the 'Locked Rate', the borrower will benefit from the lower of the current advertised Fixed Rate and the 'Locked Rate'. The benefit granted to the user to receive the lower rate at the time of settlement can be thought of as an option, specifically, the bank sells the customer a European 'at the money spot' Interest Rate swap option. The cost of this option can be calculated and should be charged by the treasury back to the line of business as an internal cost.

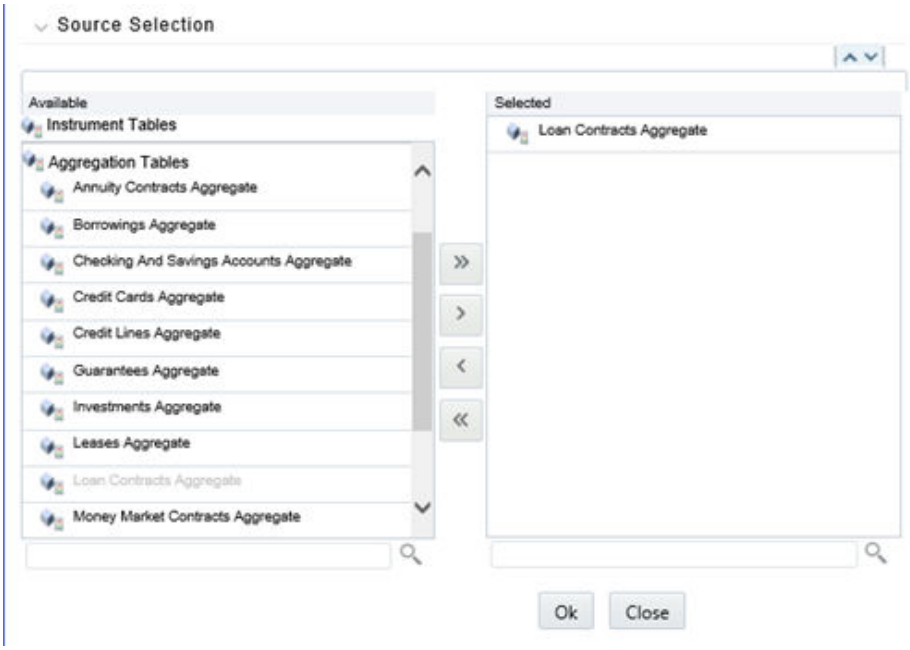
Oracle FTP provides the capability to calculate the 'rate lock' option cost. The general approach assumes that loan commitment information will be available in sufficient detail from the source systems to support cash flow transfer pricing using forward FTP curves and all required information describing the terms of the Rate Lock.

32.2 Transfer Price using Forward Rates and Calculate Rate Lock Option Costs

The Standard FTP Process provides setup options that allow you to Transfer Price Data in the Loan Commitments table using Forward Rates and calculate the Related Rate Lock option costs. To do this, please select the Loan Commitment table under Source Selection.

The following options are available on the Calculation Selection page.

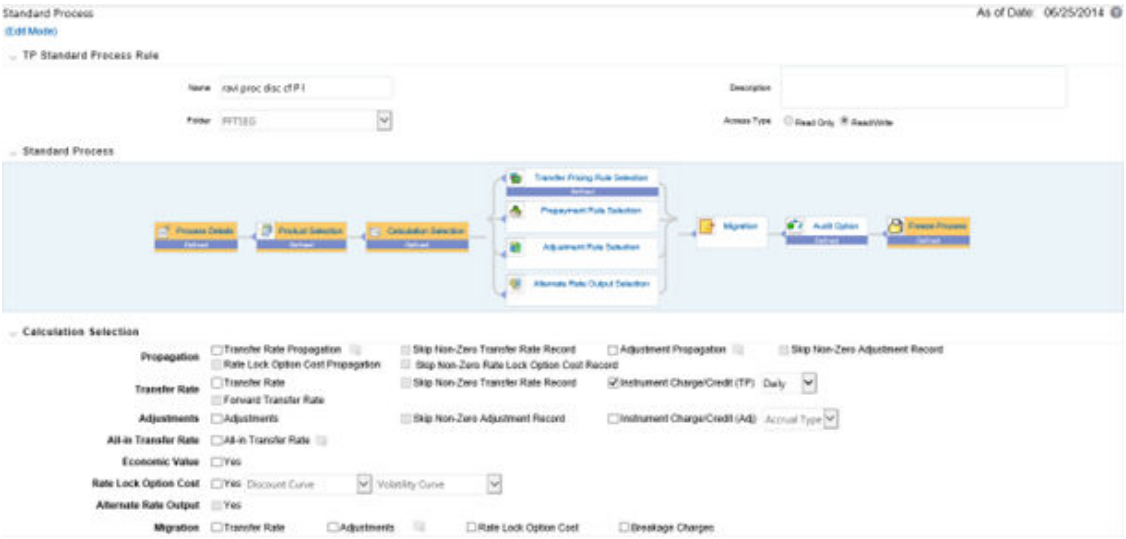
Figure 32-1 Source Selection



Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> These calculation options assume the user has also selected the Loan Commitments table on the Product Selection page under Source Selection:
>
> - **Forward Transfer Rate**: The Forward Transfer Rate option is required when the user wants to use an Implied Forward Curve as the reference curve for Transfer Pricing a future dated instrument records.
> - **Rate Lock Option Cost**: The Rate Lock Option Cost selection option along with drop-down lists for selecting the Risk-free Curve and Discount Curve are required for calculating the related Rate Lock Option Costs.
> - **Migration of Rate Lock Option Costs to Ledger Stat**: If the Rate Lock Option Cost to Ledger Stat migration is needed, users should additionally select this option.

Figure 32-2 TP Standard Process Rule



32.3 Rate Management

As shown in the TP Process, the Rate Lock Option Cost calculation requires two inputs both of which come from the Rate Management > Interest Rates page.

- **Discount Curve**: This can be a standard Interest Rate Curve.
- **Volatility Curve**: This is a special form of Interest Rate Curve, where the volatility curve option has been selected.

To set up a volatility curve, while defining a new Interest Rate Curve, select the checkbox – 'Volatility Curve'. This checkbox selection signifies that the curve will hold volatility rates. Select the Reference Currency of the IRC being created.

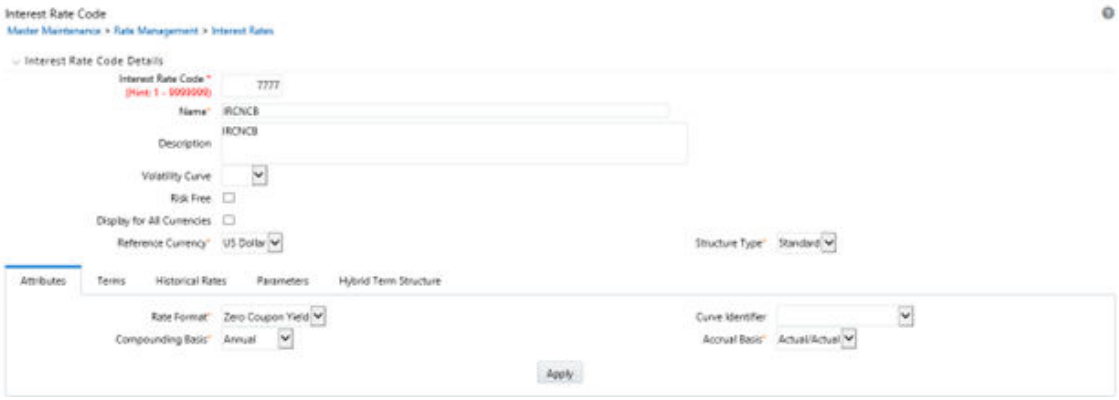
When 'Volatility Curve' is checked, the following fields are disabled and are not applicable:

- Compounding basis

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Accrual basis
- Rate Format

Figure 32-3 Interest Rate Code



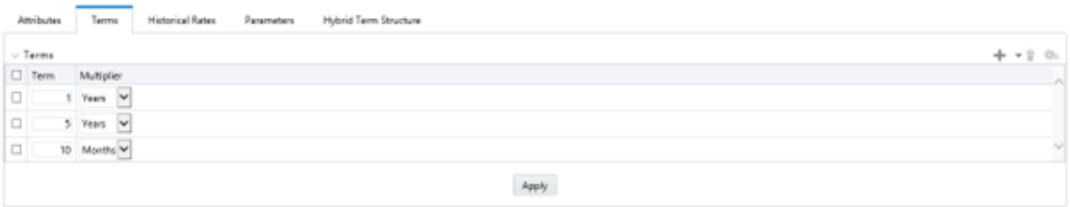
Terms tab: For a Volatility Curve, the Terms tab displays two types of terms – The Contract Term (Loan Term) and the Expiration Term (Rate Lock (option expiry) Term). Users must provide the volatility inputs for all combinations of Contract Term and Expiration Term.

> **Note**
>
> In the moneyness dimension, associated with option volatility is not required, because Rate Lock Options are assumed to be granted at the money.

The following steps are required to complete the setup of a volatility curve:

- **Terms tab - Contract Term**: Add rows and input terms for the number of required Loan Terms. These are the maturity term of the loan. Select **APPLY** to save the data.

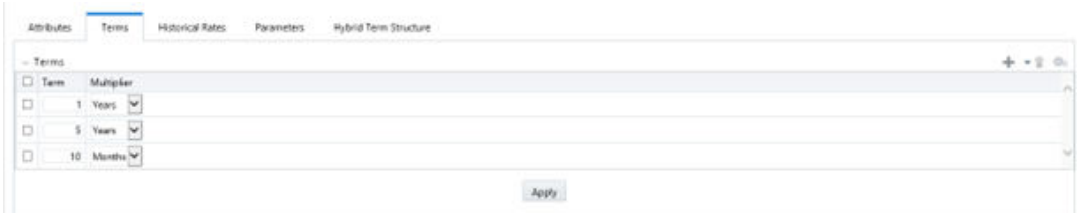
Figure 32-4 Terms Tab – Contract Term



- **Terms Tab – Expiration Term**: Add rows and Input Terms for the number of required Expiration Terms. These correspond to the number of Rate Lock Terms offered. Select **APPLY** to save the data.

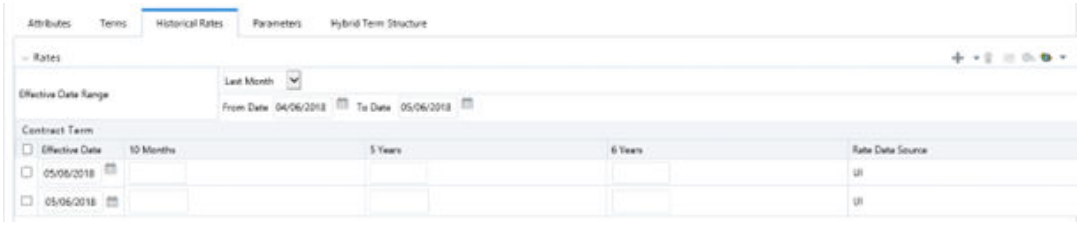
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 32-5 Terms Tab – Expiration Term



- **Historical Rates tab**: After defining and applying the volatility curve dimensions, navigate to the Historical Rates tab and input the volatility rates for each combination of the loan term and rate lock term and for each effective date that you wish to store historical volatility data.

Figure 32-6 Historical Rates tab



Select **Apply** when finished to save the data.

32.4 Implied Forward Rates Output

Users have the option to output the implied forward rates to an Audit Table if needed for validating the Transfer Rate result. To support the validation requirement, the FTP engine (optionally) writes the forward rates to an Audit table - FSI_O_FTP_IMP_FWD_RATES_AUDIT. The option to write implied Forward Rates is available in the Standard TP Process on the Audit block.

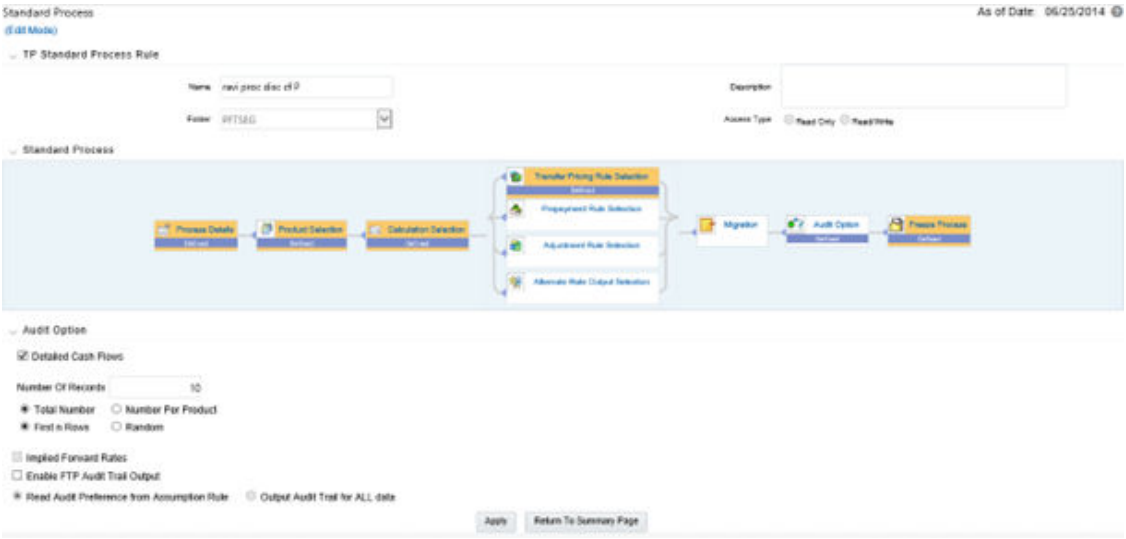
The Implied Forward Rates audit output is linked to the Detailed Cash Flows selection and Number of Records selection as a way to limit the amount of Forward Rate Output. During processing, the FTP engine will output Forward Rates for any instrument records that also have detailed cash flows output, which is one forward curve corresponding to each instrument record. We assume that users will only need to output implied Forward Rates for a limited set of data records to validate the Forward Curves and the related Transfer Rates generated by the TP engine.

The checkbox Implied Forward Rates will be active only when:

- The Forward Transfer Rate option is selected on the Calculation Selection block.
- The Detailed Cash Flows checkbox is selected in the Audit block.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 32-7 Standard Process



32.5 Transfer Pricing Future Dated Instruments (Loan Commitments)

The base FTP rate quoted on the Commitment Start date is calculated using the current Forward Rates corresponding to the tenor of the Rate Lock period requested and the loan period/payment frequency based on the TP method selected. If an instrument record has a commit_start_date <= as_of_date and origination_date > as_of_date, then FTP will generate implied forward rates based on the commit_start_date spot IRC curve.

Table 32-1 Implied Forward Rate

Effective Date	Lookup Tenor
COMMIT_END_DATE (for example, 30, 60, 90 days forward)	Term of the loan (or) Related cash flow terms, depending on the TP method

After the record is identified as a Rate lock record, the Forward Curve is generated for the same term points as the Spot curve (selected in the Transfer Price Rule UI). After this is done, the TP method that is selected follows its usual flow, except for using the Forward TP IRC instead of the Spot TP IRC. If required tenors are not found, Interpolation is used.

Within the Loan Commitment table, prior day results can be propagated forward, so post-processing, all open commitments will have a Transfer Rate and Rate Lock Option Cost assigned through either propagation or new calculation.

> **Note**
>
> The FTP engine uses only the Standard Term for Forward Rate based TP calculation and not the Remaining Term. Commitment Term and Commitment Term Mult are mandatory attributes for forward starting instruments as it will help to calculate the Implied Forward Rates which will be used for the Rate Lock Option Cost calculations on Committed Instruments.

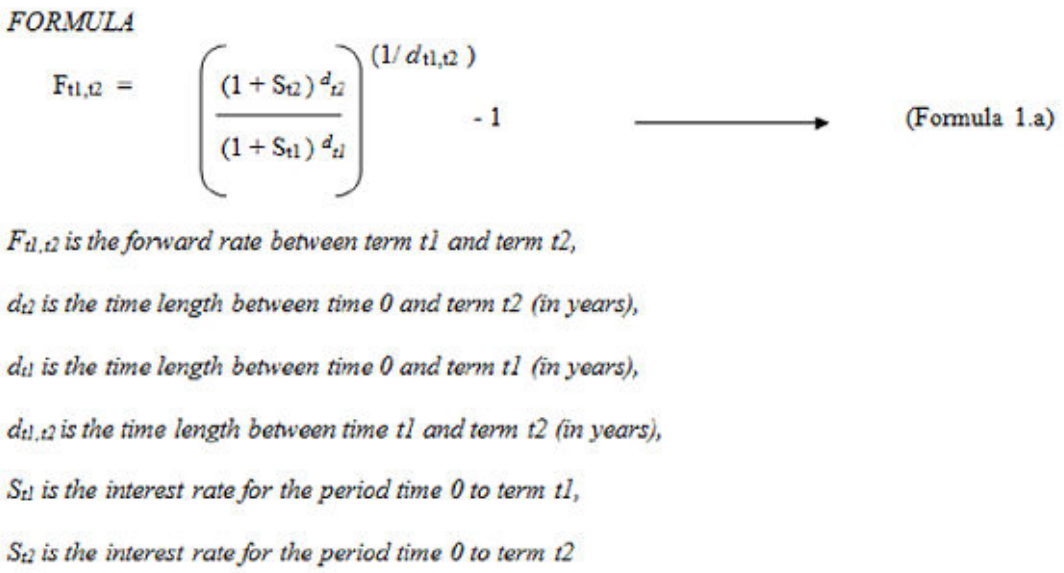
Copyright © 2011, 2026, Oracle and/or its affiliates.

- Implied Forward Rate Calculation
  An Implied Forward is that rate of interest that is predicted to be the spot rate in the future.
- Option Cost Calculation
  The Option Cost is calculated using the following formula:

32.5.1 Implied Forward Rate Calculation

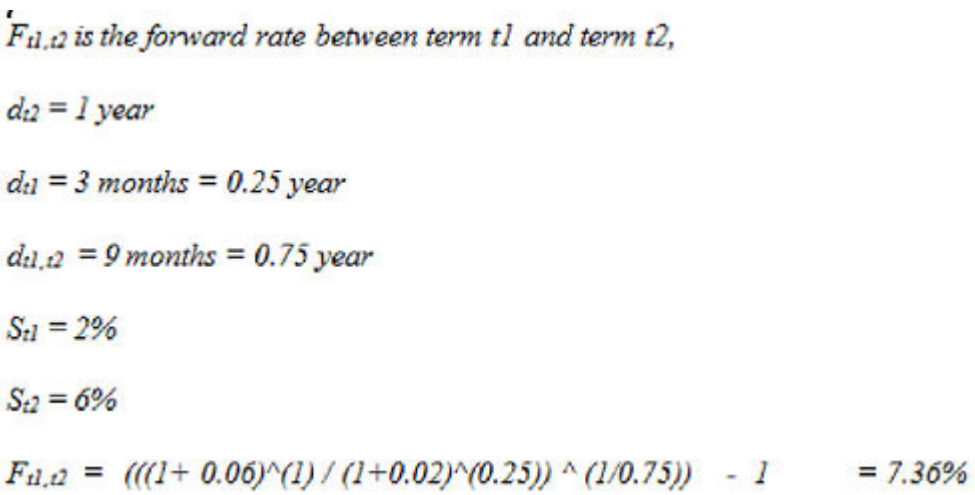
An Implied Forward is that rate of interest that is predicted to be the spot rate in the future.

Figure 32-8 Implied Forward Rate Calculation Formula



If 1 year TP Rate is 6.00% and 3 month TP Rate is 2.00% we can calculate the 3 months forward implied 9-month rate as follows:

Figure 32-9 3-month Forward Implied 9-month Rate Calculation Formula



Copyright © 2011, 2026, Oracle and/or its affiliates.

Therefore, the market is implying that in 3 months, 9 month TP Rate will be 7.36%.

Rate Lock Option Cost Calculation: The Rate Lock Option Cost calculation uses a standard Black European swap pricing formula. This calculation is triggered by a Standard FTP Process and can be performed for both fixed-rate and adjustable-rate instruments. The following conditions must hold true for instrument records in the FSI_D_LOAN_COMMITMENTS table:

```
commit_start_date <= as_of_date

origination_date > as_of_date
```

Figure 32-10 Black Formula for calculating Rate Lock Option Cost

$$d1 = \frac{Log(F/X) + T*0.5v^2}{Sqr(T)*v}$$

$$d2 = \frac{Log(F/X) + T*0.5v^2}{Sqr(T)*v} - Sqr(T)*v = \frac{Log(F/X) - T*0.5v^2}{Sqr(T)*v}$$

$$OptionCost = \frac{1-(1+F/m)^{(-t1*m)}}{F} * Exp(-r*T)*[F*CND(d1) - X*CND(d2)]$$

where:

- *t1 is the term to maturity of the loan*
- *T is the term to expiry of the option*
- *F is the forward instrument rate – to be picked from the calculated Forward curve*
- *X is the strike rate (same as F – Forward Instrument rate)*
- *v is volatility*
- *r is the continuously compounded "risk free" rate to option expiry*
- *m is the payment frequency of the underlying swap.*

Calculate Rate Lock Option Cost for a Rate Lock arrangement with the following details:

Table 32-2 Example: Option Cost Calculation

Loan Face value - ORG_BOOK_BAL	10,000,000	
Tenor of Loan - ORG_TERM & ORG_TERM_MULT	5	Years
Locked TP Rate - TRANSFER_RATE	8.20%	
Rate Lock Commitment period - COMMIT_TERM & COMMIT_TERM_MULT	90	days
Loan Payment frequency - PMT_FREQ	6	Months
Volatility	20%	
Risk-free rate to Option Expiry	4%	

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 32-3 Required Inputs

Term to maturity of the loan	t1	5	years
Term to the expiry of the rate lock option	T	0.2465753	years
Strike rate - Locked TP Rate (Forward TP Rate as on Loan Origination)	X	8.20%	
Volatility			
Details for Volatility - From the historical volatility curve that is loaded in Rate Management by the user, pick Volatility% with			
EFFECTIVE DATE = COMMIT_START_DATE and LOOKUP TENOR = Tenor of the Loan. In Release 6.0, 2 Dimensional Volatility curve was introduced with Contract term and Expiry term as the 2 dimensions.	v	20%	
Payment frequency of the loan	m	6	months
Continuously Compounded TP rate to option expiry	r	4.08%	(See the calculation (1) below)
Implied Forward TP rate	F	8.20%	(See calculation (2) below)

Table 32-4 Intermediate Calculations

(1) Continuously Compounded TP rate to option expiry	r	4.08%	
(2) Implied Forward TP rate			
(FDD v1.1 - Implied Forward Rate Calculation - Section 6.1.2.1)			
Inputs required - (Terminology for these inputs is according to Section 6.1.2.1)	F		
dt1 - Commitment term of Rate Lock	0.246575	years	
dt1,t2 - Tenor of Instrument	5	years	
dt2- Time length between Commitment Start Date and Loan maturity	5.246575	Years	
St1- Spot Interest Rate as on COMMIT_START_DATE for Commitment Term of the Rate Lock (COMMIT_MAT_DATE – COMMIT_START_DATE)	4%		

Copyright © 2011, 2026, Oracle and/or its affiliates.

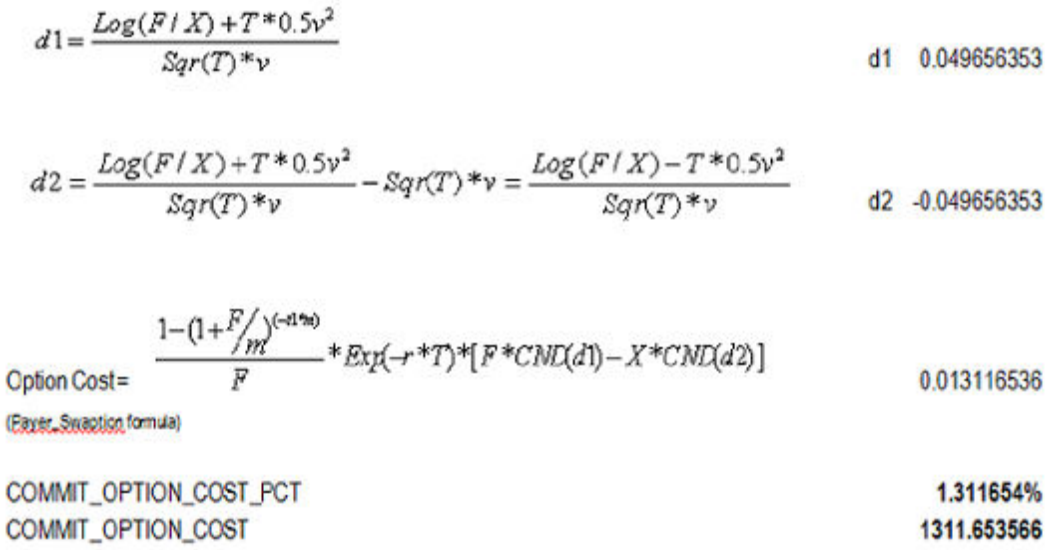
Table 32-4 (Cont.) Intermediate Calculations

St2- Spot Interest Rate as on COMMIT_START_DATE for Time length between Commitment Start Date and Loan maturity	8%	
Implied Forward Rate, F (Formula given above in explanation)	0.0820119	8.20%

32.5.2 Option Cost Calculation

The Option Cost is calculated using the following formula:

Figure 32-11 Option Cost Calculation Formula

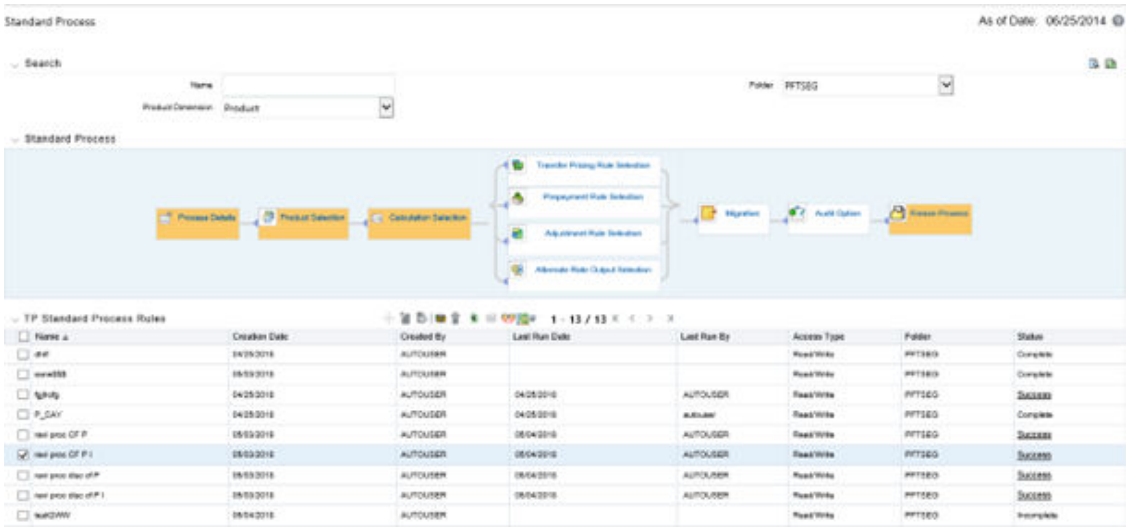


32.6 Execution and Results

You can execute the TP Process to calculate Forward Rates and Rate Lock Option Costs by selecting the process in the summary screen and then click **Run**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 32-12 Standard Process



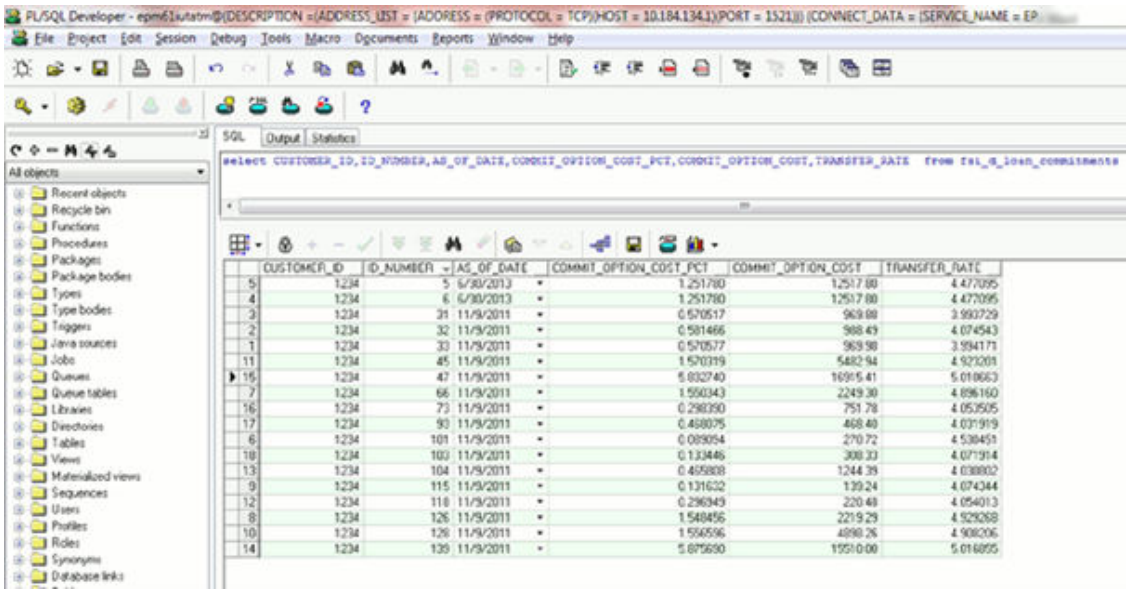
The following are the relevant output columns related to this feature:

- COMMIT_OPTION_COST_PCT: Rate Lock Option Cost in Percent terms
- COMMIT_OPTION_COST: Rate Lock Option Cost in Amount terms
- TRANSFER_RATE: Forward Transfer Rate

You can output the implied forward rates to an Audit Table if needed for validating the Transfer Rate result. To support the validation requirement, the FTP engine (optionally) writes the forward rates to an Audit table - FSI_O_FTP_IMP_FWD_RATES_AUDIT.

Sample results of SQL queries shown below:

Figure 32-13 Sample SQL Query Results



Copyright © 2011, 2026, Oracle and/or its affiliates.

33
FTP Rate Card

Rate card functionality allows the users to select standard products for viewing in their daily FTP Rate report.

Administrators schedule a daily FTP run for the selected set of standard products and end users can view daily rates for relevant standard products by defining their Daily rate card reports. Rate cards are used for informational purposes only. They are used to give an idea of the prevailing transfer rates as per the current market scenario before actual transactions take place, unlike a Standard Transfer Pricing Process where a bank runs at the end of each month to generate Transfer Rates for each booked instrument record.

To understand the use case better, think of a situation where a user (Bank Personnel, Account officer, so on) wants to refer to the current cost of funds rate before the transaction rate is quoted. Having access to this up-to-date information allows the banker to be pro-active with their pricing decisions having full knowledge of the cost of funds and prospective rate spread before quoting a rate to the customer.

Topics:

- Setting up a Product
- Rate Report Templates
- Defining a Rate Report
- Viewing Rate Card Report
- Setting up a Product
  Product setup allows Administrators to define the default Product Characteristics for standard products. The Administrator will define these assumptions for Products during the application setup through the provided user interface.
- Setting up Batch Execution of Rate Card
  For the defined Products to become “dummy” instrument records and for these records to be Transfer Priced, a Batch process is required. The following steps describe how to create the required batch process.
- Rate Report Templates
  Rate Card Report templates help you to create a customized Rate Card Report by choosing the columns you want to see in your report. These columns can include/exclude standard out of the box columns as well as other columns that are properly registered. Following is the updated flow with an extra (new) step of defining column templates, while defining Rate Card Report:
- Defining a Rate Report
  The Rate Card window allows users to select the set of standard products, report templates and the relevant historical date range to include in their report.
- Viewing Rate Card Report
  Rate Card report is available upon successful completion of the Pricing Run.

Copyright © 2011, 2026, Oracle and/or its affiliates.

33.1 Setting up a Product

Product setup allows Administrators to define the default Product Characteristics for standard products. The Administrator will define these assumptions for Products during the application setup through the provided user interface.

The Products window shows the list of default Product Characteristics for standard products and you can define the properties for them.

The procedure for working with and managing Products is similar to that of other Oracle Funds Transfer Pricing business rules. It includes the following steps:

- Searching for Products
- Creating a Product Definition
- Viewing and Editing Product Definitions
- Copying Product Definitions
- Deleting Product Definitions

As part of creating and editing Product definitions, the user defines the properties for applicable products.

Figure 33-1 Products screen



- Defining a Product
  To define a product:
- Defining a Product: An Example
  To define a product:

33.1.1 Defining a Product

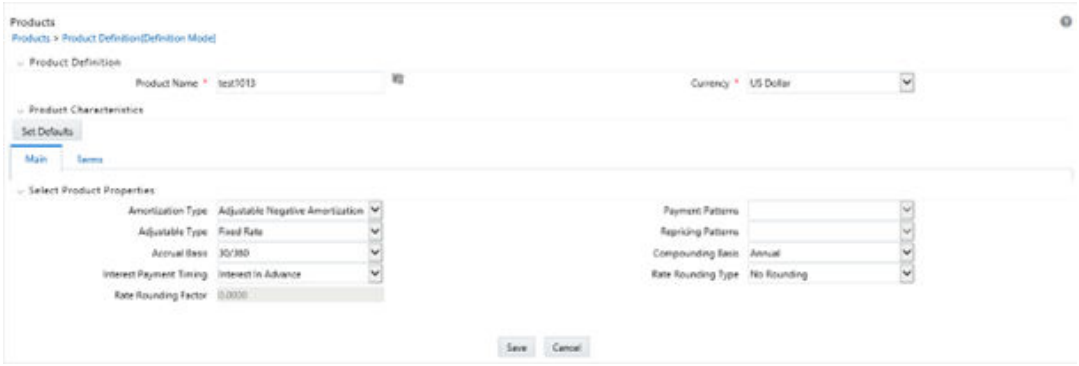
To define a product:

1. Navigate to **FTP Rate Card**, and select **Products** to access the Products window.
2. Click the **Add** icon to create a new product definition.
   The definition of Product is a part of the Create or Edit Product Definition page. When you click Apply or Save on the Create Product Definition page, the product definition is saved and the Product will be displayed on the Product summary page.

> **Note**
>
> Only those Products, which are defined by the Administrator, will be included in the daily FTP Rate Card Process and similarly, only defined products will be available to end-users for Rate Card reporting.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-2 Product Definition screen



Prerequisites: Performing basic steps for creating or editing a Product.

Procedure: This table describes the key terms used for this Procedure.

Table 33-1 Fields and Descriptions from the Products summary page

Term	Definition
Product	Provides a list of the leaf dimension members for the Product dimension selected in Application Preferences.
Currency	The corresponding currency of the instrument to be priced.
Set Defaults	Select the Set Defaults option, to restore default Product Characteristics.
Main Tab	
Amortization Type	Select the Amortization Type. This defines the method by which an account's principal and interest will be Amortized. The Default Amortization Type is Non Amortizing.
Payment Patterns	Optionally, select the Payment Pattern. This list is defined through the Payment Pattern user interface.
Adjustment Type	Select the Adjustment Type. This selection indicates if the product is a fixed-rate or adjustable-rate. The Default Adjustment Type is Fixed Rate.
Repricing Patterns	Optionally, select the Repricing Pattern. This list is defined through the Repricing Pattern user interface.
Accrual Basis	Select the Accrual Basis. The interest accrual is calculated on this basis. The default value is Actual/Actual.
Compounding Basis	Select the Compounding basis. This selection indicates the compounding frequency used to calculate the interest income. The compounding basis for the interest payments can be monthly, annually, simple, and so on. The default value is Simple.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 33-1 (Cont.) Fields and Descriptions from the Products summary page

Term	Definition
Interest Payment Timing	Define the Interest Payment Timing. You can pay the interest in Advance, Arrears. The default value is Interest in Arrears.
Rate Rounding Type	Select the Rate Rounding Type to round off the Interest Rate. This selection indicates how the rate assigned to the product will be rounded. The default value is No Rounding.
Rate Rounding Factor	Enter the rate-rounding factor. If the Rate Rounding Type is Round Up, Round Down, or Round Nearest, then the rate-rounding factor determines the precision of the rounding. The possible range of values for this is 0.0000 – 9.9999. The default value is 0.0000. This option is not applicable if Rate Rounding Type is selected as No Rounding.
Term Tab	
Original Term	Enter the Original Term to define the contractual term from the origination date. Note that it is possible to define more than one term for the selected product/currency. If more than one term is defined, then multiple records are created for pricing, i.e. one corresponding to each Product/ Currency and Term.
Payment Frequency	Enter the payment frequency. This allows you to define the frequency of payment.
Repricing Frequency	Enter the Repricing Frequency to define the frequency of rate change of a product.
Amortization Term	Define the assumed term used for payment calculation purposes. This will be equal to the Original Term of the instrument. It should only be different in cases where the instrument does not fully Amortize over the life of the product. i.e. there is a lump sum payment due on the maturity date.
Interest Rate Code	Enter the Interest Rate Code to be used for finding the coupon rate on the product.
Margin	Enter the margin that is the contractual spread, which is added to the pricing index and results in the financial institution's retention (net) rate.
Rate Set Lag	Define the period by which repricing lags the current interest rate changes. The default value is 0 Months.
Tease Period	Define the Tease Date, that is when the Tease Rate (introductory rate) ends and the normal product rate begins. The default value is 0 Months. This entry is disabled, if the Adjustable Type (defined from Main Tab) is a Repricing pattern or fixed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 33-1 (Cont.) Fields and Descriptions from the Products summary page

Term	Definition
Tease Discount	Enter the Tease Discount. The default value for tease discount is 0.0000. This entry is disabled if the Adjustable Type is a repricing pattern or fixed rate or if the teasing period is zero.
Commitment Start Date	The commitment start date defaults to the system date.
Commitment Term	Enter the commitment period term in Days. Based on this input the commitment end date is calculated.
Commitment End Date	The commitment end date is calculated automatically based on the commitment start date and commitment term.

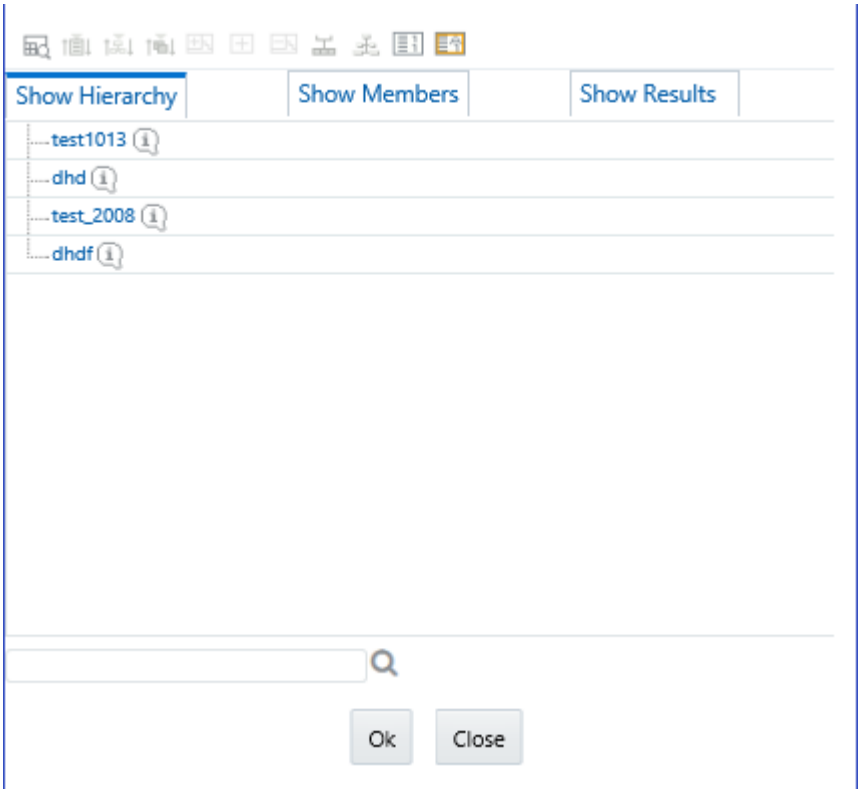
<Enter a single subject here.>

33.1.2 Defining a Product: An Example

To define a product:

1. Select a standard Product using the hierarchy Browser.

Figure 33-3 Hierarchy screen



2. Select the corresponding currency.
3. Define the properties of the Product in the Main tab.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-4 Product Definition – Main Tab



4. Define the properties of the Product in the Terms tab.

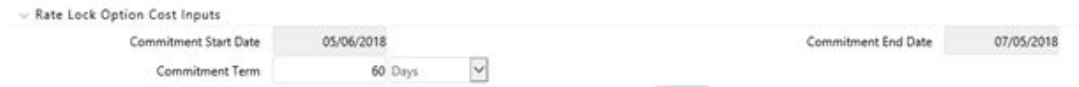
Figure 33-5 Product Definition – Terms Tab



> **Note**
>
> Under a single product/currency selection, users may define multiple combinations of Original Term, Payment Frequency, and Repricing Frequency. This is done by adding records using the "+" icon. Multiple records are useful when generating daily pricing sheets, where the standard product has multiple terms. For example, the product could be "Term Deposits", but users may want to view the daily FTP Rates for each available term, for example, 3 months, 6 months, 12 months, 18 months, and 60 months. This example is possible by adding and defining 5 records under a single product/currency selection. For more information on the meaning of each of these fields, see the OFS Cash Flow Engine Reference Guide.

5. Check Rate Lock Option Cost Inputs to set up and calculate Rate Lock Option Costs.

Figure 33-6 Rate Lock Option Cost Inputs



6. Click Save.

> **Note**
>
> On save of each Product (Main and Terms), one or more record(s) will be inserted in fsi_m_prod_characteristics table. When Multiple terms are defined for each product, multiple entries will be inserted into the fsi_m_prod_characteristics table. After the completion of the first task tpol.sh run, fsi_pm_generated_instrmts table is populated.

Copyright © 2011, 2026, Oracle and/or its affiliates.

33.2 Setting up Batch Execution of Rate Card

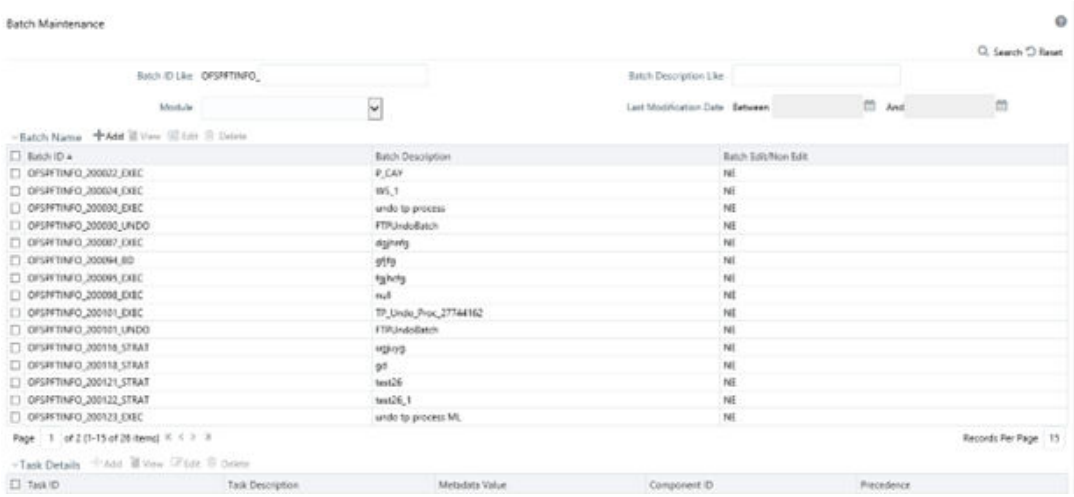
For the defined Products to become “dummy” instrument records and for these records to be Transfer Priced, a Batch process is required. The following steps describe how to create the required batch process.

Create a Batch for running the daily FTP process against standard products. This process should be scheduled to Run each day (or at any required frequency) by the Administrator after the daily FTP reference rates are loaded.

To set up batch execution of rate card:

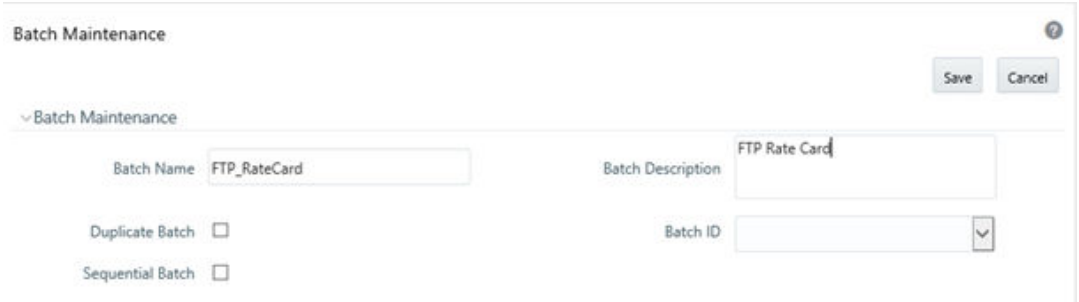
1. Click the **Batch Maintenance** option under **Operations**.

Figure 33-7 Batch Maintenance screen



2. Click the **Add** icon to create a Batch for running the daily FTP process against standard products.
3. Enter the Batch Name and description. Select the Duplicate Batch and/or Sequential Batch option(s) if required. Click Save to save Batch details.

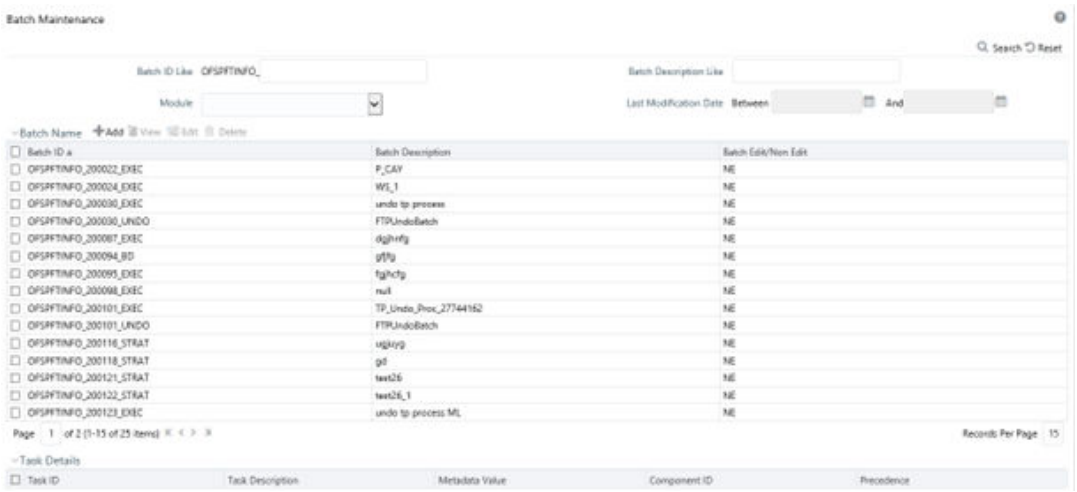
Figure 33-8 Batch Maintenance Fields



4. Search the created Batch using the Search criteria if required. Searched Batch is displayed. Select the Batch to add the corresponding tasks to it.

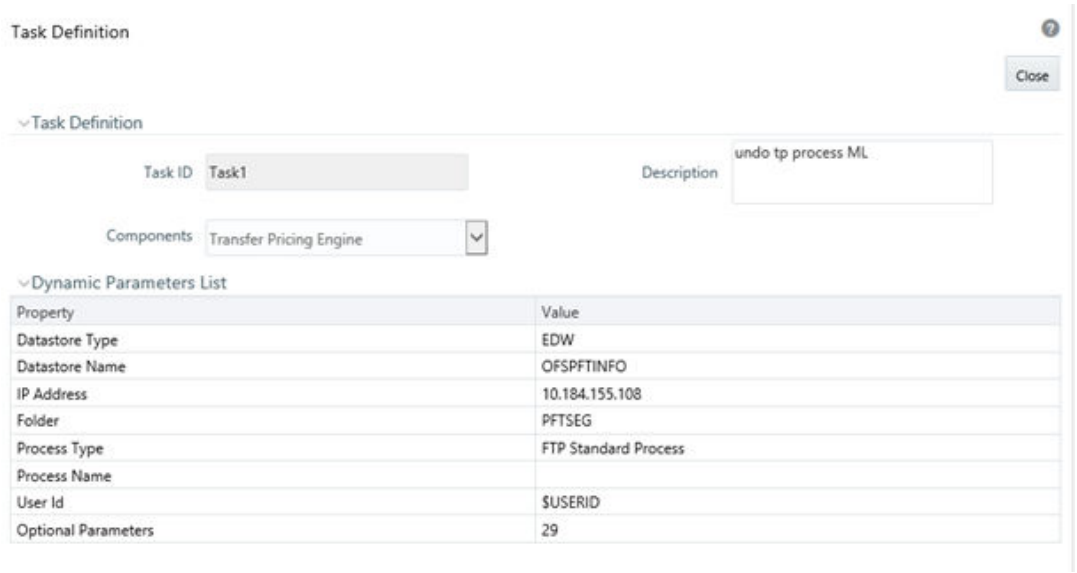
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-9 Batch Maintenance summary page



5. Click **Add** to add the first task to the Batch and enter the task description for the first Task in the Task Definition box. Select the component "RUN EXECUTABLE".

Figure 33-10 Batch Maintenance – Task Definition



6. Enter the required parameters as follows:

Table 33-2 Fields and Descriptions from the Task Definition page

Term	Description
Datastore Type	Leave the default value of EDW.
Datastore Name	Leave the default value.
IP Address	Leave the default value.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 33-2 (Cont.) Fields and Descriptions from the Task Definition page

Term	Description
Executable	Input appropriate values for the following required parameters, tpol.sh, Product dimension number, System ID number for the TP process (for example, tpol.sh,3,300052) tpol.sh is the executable file for the first task. It takes process id and product dimension type as the input parameters. tpol.sh internally calls a java program that takes data from the UI tables and populates the FSI_PM_GENERATED_INSTRMTS table. Process_id parameter usage in the tpol.sh task: The Process ID is used to identify the instrument table (i.e. the PM Generated Instruments table) and fires a delete on this table before processing. This process will work only for the FSI_PM_GENERATED_INSTRMTS table, as it uses an explicit request_id which is null. This column is available only in the FSI_PM_GENERATED_INSTRMTS table. While generating the “dummy” instrument records, the tpol.sh process picks up data from the Product Characteristic table but also joins to the process. It generates the instruments only if a TP rule is attached to the Process.
Wait	Y
Batch Parameter	Y

The name of the executable file is tpol.sh. The following Product Dimension types are supported here:

- 2 – General Ledger
- 3- Common COA
- 4- Product

7. Click **Add** to add the second task to the selected Batch and enter the description for the second task on the Task Definition page. Select the component of the task as "Transfer Pricing Engine". This process will execute a TP engine Run, reading the instrument data from the FSI_PM_GENERATED_INSTRMTS table and will update the Transfer Rate column with the calculated result.

> **Note**
>
> Users must first create a Standard TP Process for selection in this batch. The TP Process should include the “PM Generated Instruments” table under Source Selection. The calculated items should include all relevant rate calculations, such as Transfer Rate and Adjustment Rates.

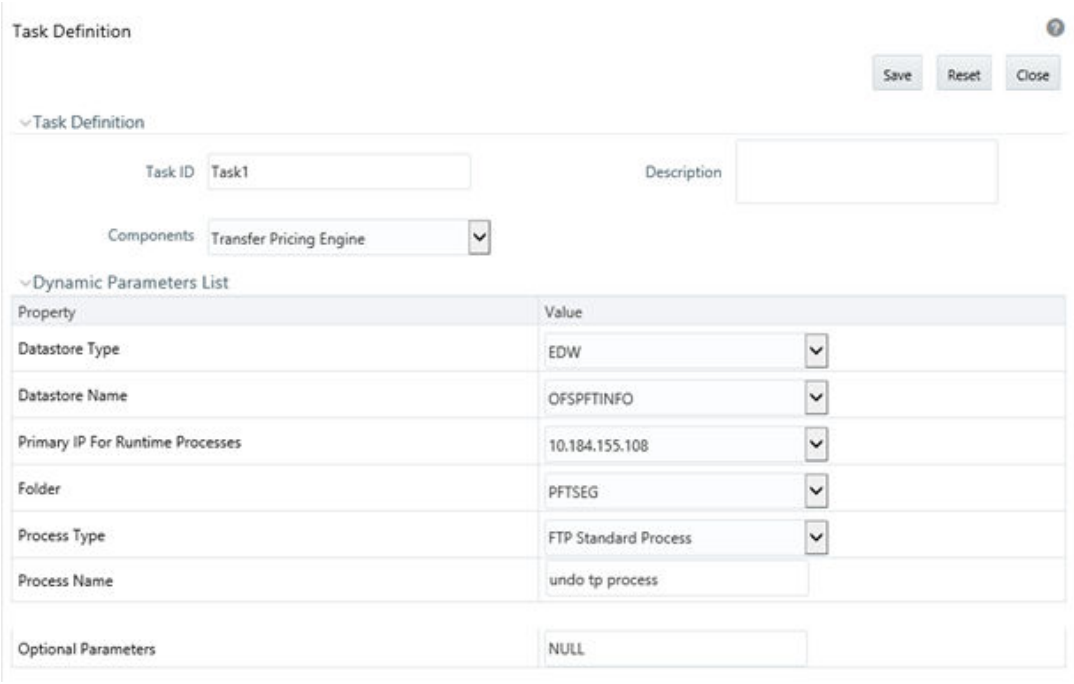
8. Enter the required parameters as follows:
   - **Folder**: Select the folder in which the definition of the process that has to be scheduled for execution is lying.
   - **Process Type**: The Process type can be Standard or Stochastic.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Process Name**: The Process Name should be defined for process metadata value (system ID of the process) which is defined in Task 1 under Executable parameter.
- **Optional Parameters**

9. Click **Add** to add the third task to the selected Batch and enter the description for the third task on the Task Definition page. Select the component of the task as "Transform Data". This process will populate the FTP rate from the fsi_pm_generated_instrmts table to the FSI_M_PROD_TRANSFER_RATE table, which is the source table for Rate Card Reporting.

Figure 33-11 Task Definition fields



10. Enter the required parameter details as follows:
    - **Rule Name**: DT_POPULATE_PM_TRANS_RATE_T_1
    - **Parameter List**

> **Note**
>
> Third step is not required anymore, as for rate card reporting, data gets picked from FSI_M_PROD_TRANSFER_RATE earlier. But now onwards, fsi_pm_generated_instrmts itself, can retain history and used for Rate Card Reporting.

11. Define the precedence of the tasks. Task 2 should be executed after Task 1, and Task 3 should be executed after Task 2.

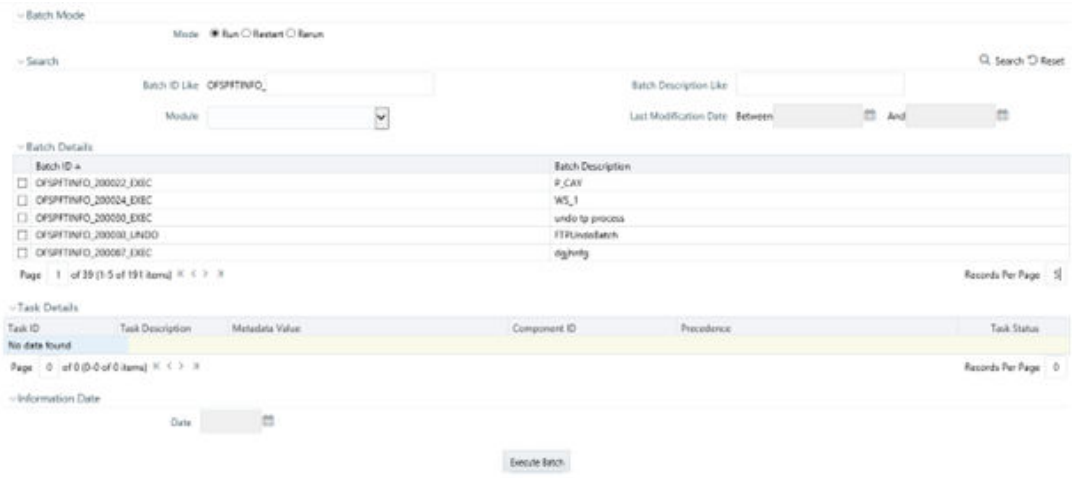
Figure 33-12 Task Details pane



Copyright © 2011, 2026, Oracle and/or its affiliates.

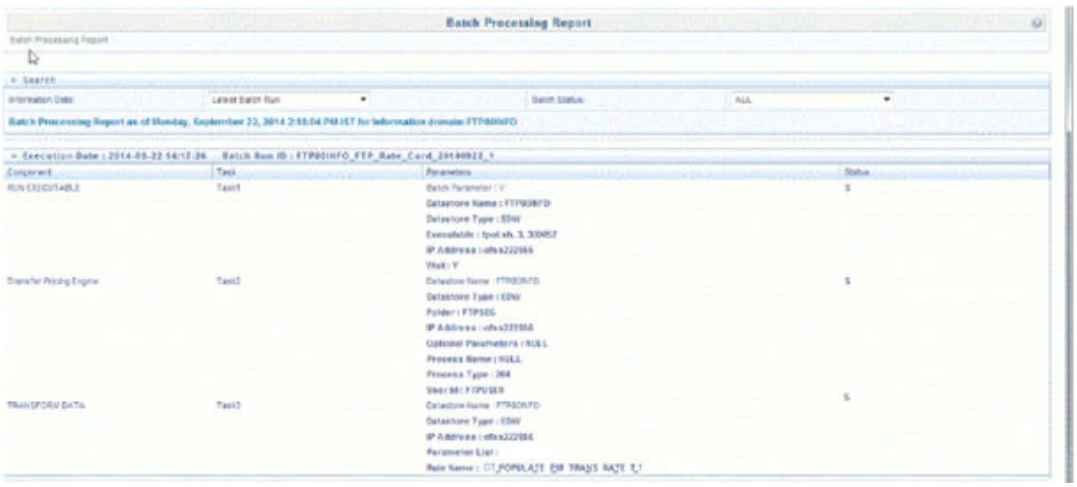
12. Click the **Batch Execution** option under **Operations** to execute the Batch. Select the Batch, which you want to execute using search criteria using the corresponding check-box.

Figure 33-13 Batch Mode pane



13. You can also use the Schedule option if you want to schedule a Batch for later or recurring processing. Click **Execute Batch**. A message will be displayed: *Batch Triggered Successfully*.
14. Click **Ok**.
15. To view the Batch Processing Report, click the **Batch Processing Report** option under **Operations**. Expand the Execution Date.

Figure 33-14 Batch Processing Report



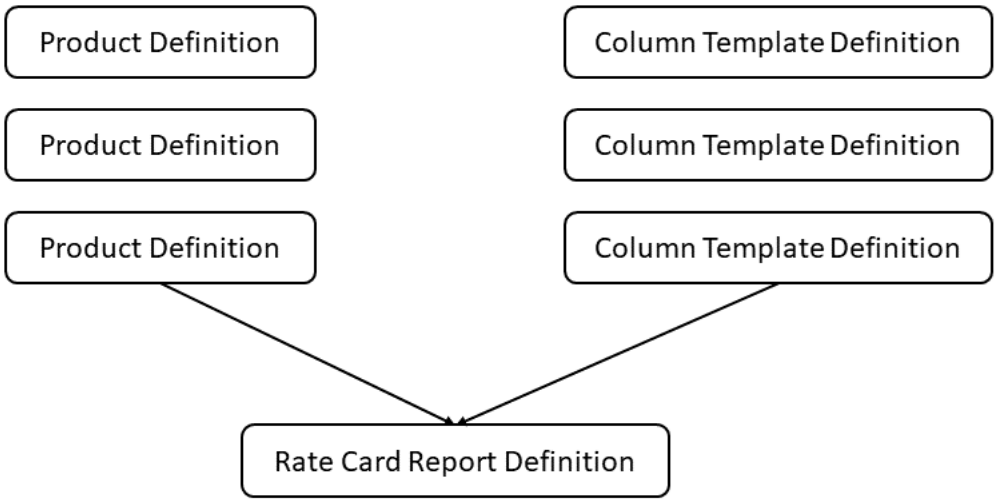
At this point, “dummy” instrument records will be generated, the records will be transfer priced, inclusive of both base TP Rates and any Add-on Rates defined through an Adjustment Rule, and the results will be moved into the Rate Card reporting table and will be available for reporting.

Copyright © 2011, 2026, Oracle and/or its affiliates.

33.3 Rate Report Templates

Rate Card Report templates help you to create a customized Rate Card Report by choosing the columns you want to see in your report. These columns can include/exclude standard out of the box columns as well as other columns that are properly registered. Following is the updated flow with an extra (new) step of defining column templates, while defining Rate Card Report:

Figure 33-15 Rate Card Report Flow



Each template contains a user specified set of product attributes selected from the master table (FSI_PM_GENERATED_INSTRMTS). Templates is an added functionality, which will allow you to change rate card reports on the Run. Each report can be defined with more than one template, which can change on the Run as per your requirement.

- Search for the Rate Report Templates
  Search for a rate report template to perform any of the following tasks:
- Creating a Rate Report Template
  Template creation is a role specific activity, only the users with the creation rights can create a new template.
- Viewing or Editing a Rate Report Template
  To view or edit an existing template, navigate to FTP Rate Card, and select Rate Report Template. Select the template that you want to edit or view. Click the Edit or View icons in the summary screen. You can remove existing columns and add new columns along with changing the column names, which appear in the Rate Card Report.
- Copying an Existing Rate Report Template
  To copy an existing template, navigate to FTP Rate Card, and then select Rate Report Template. Select the template to be copied and click the **Copy** icon in the summary screen.
- Deleting an Existing Rate Report Template
  To delete an existing template, navigate to FTP Rate Card, and then select Rate Report Template. Select the check-box against the template that you want to delete. You can select more than one template for deletion.

Copyright © 2011, 2026, Oracle and/or its affiliates.

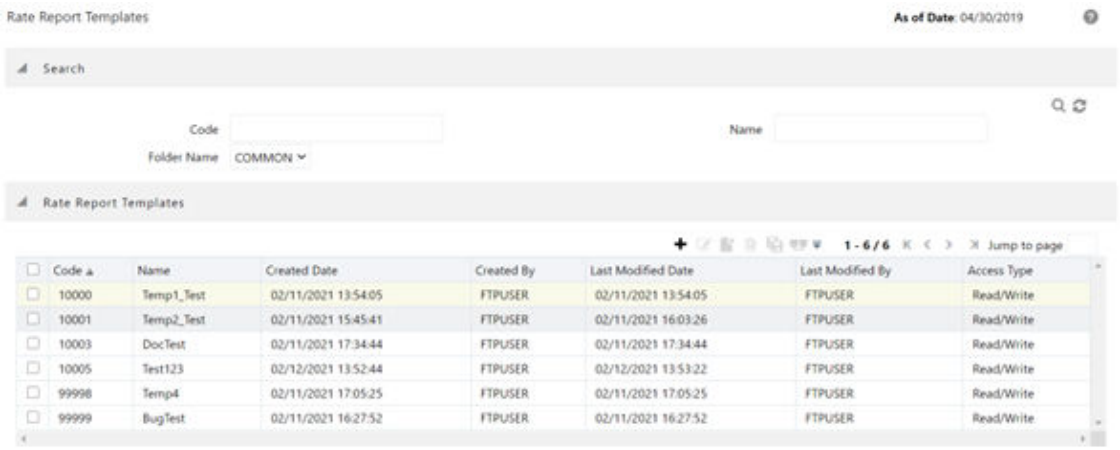
33.3.1 Search for the Rate Report Templates

Search for a rate report template to perform any of the following tasks:

- Edit
- View
- Delete
- Copy
- Check Dependencies

Defining a Rate Report Templates is a prerequisite to perform any operation on the Template, like search. To define a Rate Report Template, From the LHS menu, select Funds Transfer Pricing, select FTP Rate Card, and then Rate Report Template to display the Rate Report Templates screen.

Figure 33-16 Rate Report Templates screen



From this screen, you can either search for a template, or define a new template.

- To search for an existing template, enter the Template Code or name along with the folder Name where template is stored and click the **Search** icon. The summary window displays the details of the Templates as per the search criteria.
- Subsequently, you can select a template from the summary screen to view, edit, copy or delete the selected template.

33.3.2 Creating a Rate Report Template

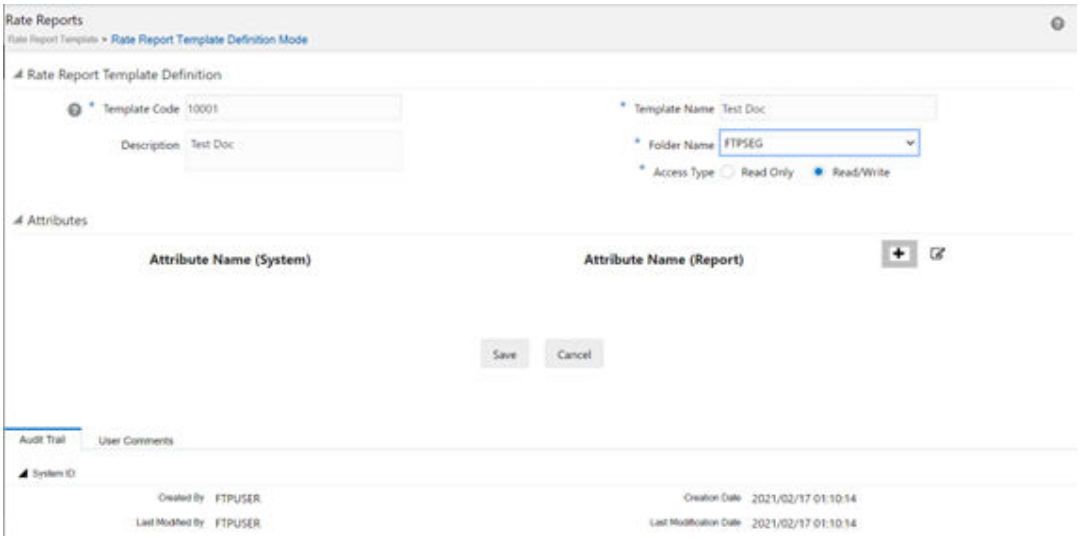
Template creation is a role specific activity, only the users with the creation rights can create a new template.

To create a Rate Report Template:

1. Navigate to Rate Report Templates screen.
2. Click the **Add** icon to add a new template.

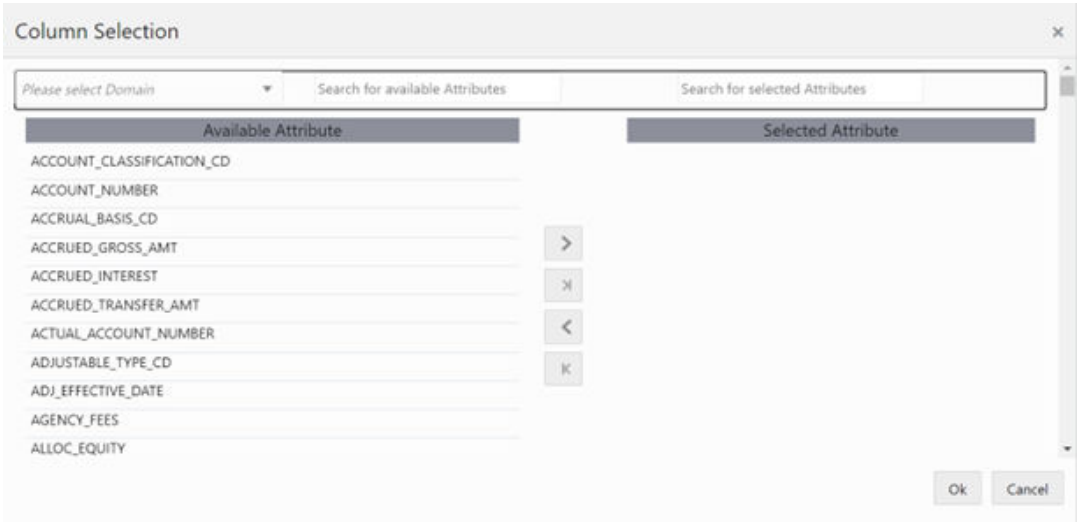
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-17 Rate Report Template Definition Mode



3. Enter the following details for the new report template:
   - **Template Code**: A unique numeric code within the range of 10000 to 99999.
   - **Template Name**: A name for the template.
   - **Description**: A description for the template.
   - **Folder Name**: A folder where this template is saved.
   - **Access Type**: The access type that you want to give to this template for other users.
4. Click the **Add** icon to display the Column Selection window where you can select the attributes to the template.

Figure 33-18 Column Selection screen



5. Select the **Domain** from which you want to select the attributes. The attributes available for the selected Domain will be displayed in the **Available Attributes** column. If no Domain is selected, this screen displays the all the Available Attributes.
6. Select the Attributes from the **Available Attribute** column and click the > icon or >| icon to move them to the **Selected Attribute** column. If you select any attributes by

Copyright © 2011, 2026, Oracle and/or its affiliates.

unintentionally, you can move them back to the Available Attribute column by clicking the < icon or |< icons.

7. Click **OK** to save the selected attributes to the report template.

33.3.3 Viewing or Editing a Rate Report Template

To view or edit an existing template, navigate to FTP Rate Card, and select Rate Report Template. Select the template that you want to edit or view. Click the Edit or View icons in the summary screen. You can remove existing columns and add new columns along with changing the column names, which appear in the Rate Card Report.

Figure 33-19 Edit an Existing Rate Report Template

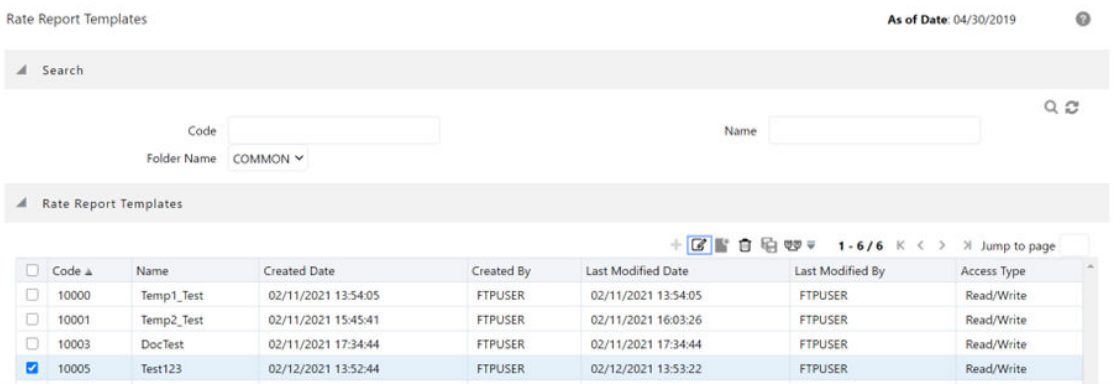
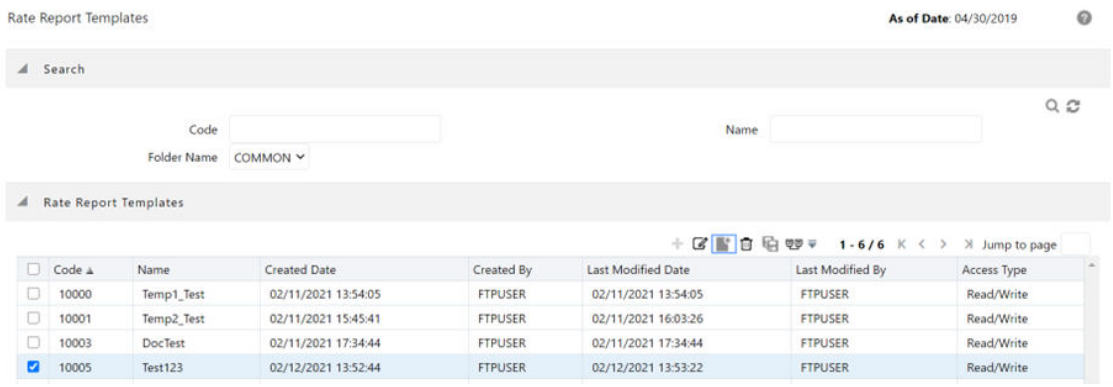


Figure 33-20 View an Existing Rate Report Template

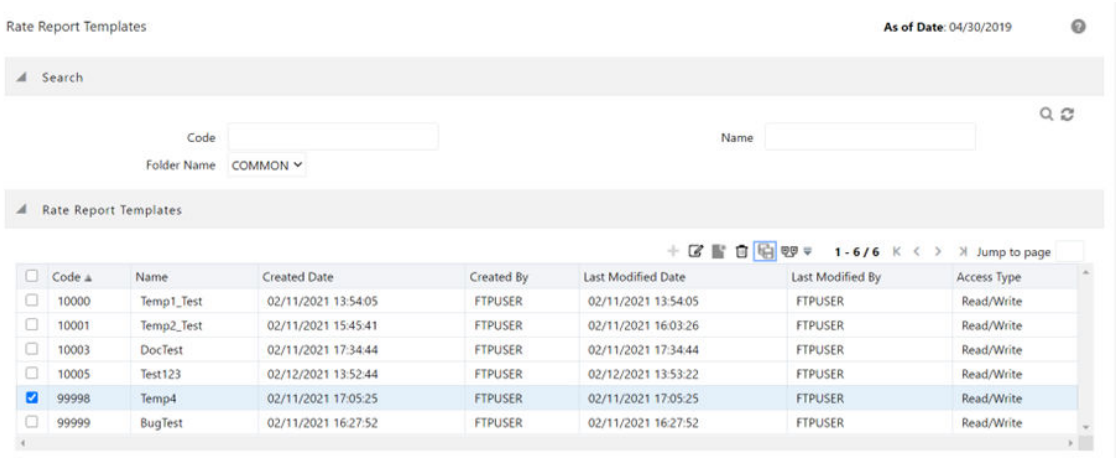


33.3.4 Copying an Existing Rate Report Template

To copy an existing template, navigate to FTP Rate Card, and then select Rate Report Template. Select the template to be copied and click the **Copy** icon in the summary screen.

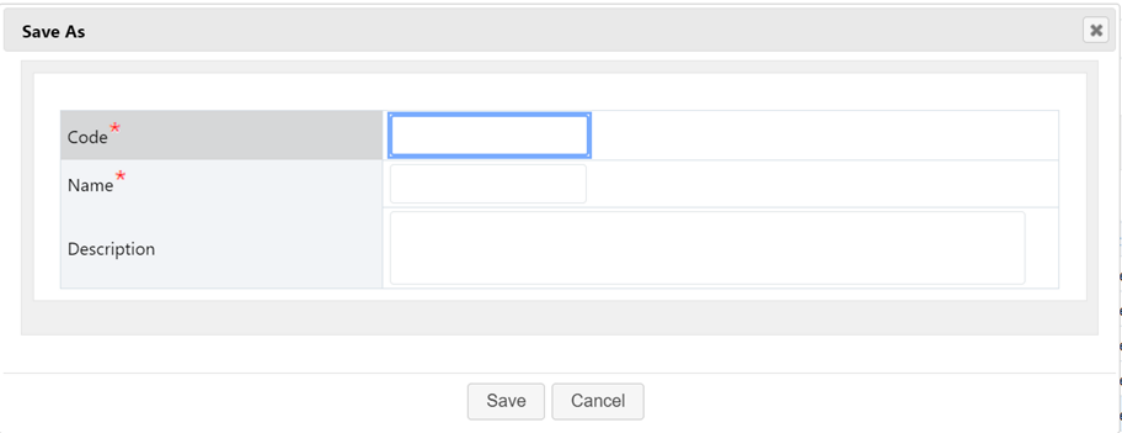
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-21 Copying an Existing Rate Report Template



When you select an existing template and click the Copy icon, the following Save As window is displayed where you can save the selected template with a new name and code.

Figure 33-22 Copying an Existing Rate Report Template – Save As window



Enter the Code, Name, and Description for the template and click Save. The new template is displayed in the summary screen.

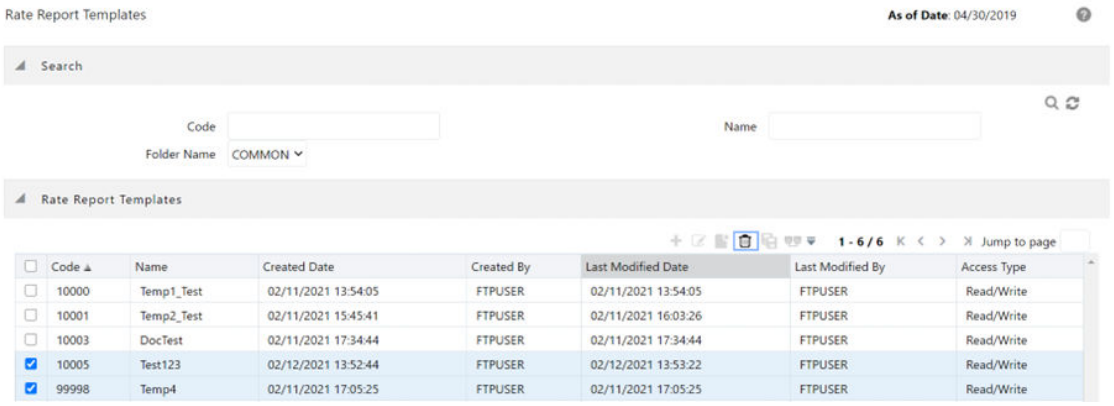
33.3.5 Deleting an Existing Rate Report Template

To delete an existing template, navigate to FTP Rate Card, and then select Rate Report Template. Select the check-box against the template that you want to delete. You can select more than one template for deletion.

Click the Delete icon in the summary screen. If there is no dependency, the selected templates are deleted with a confirmation message.

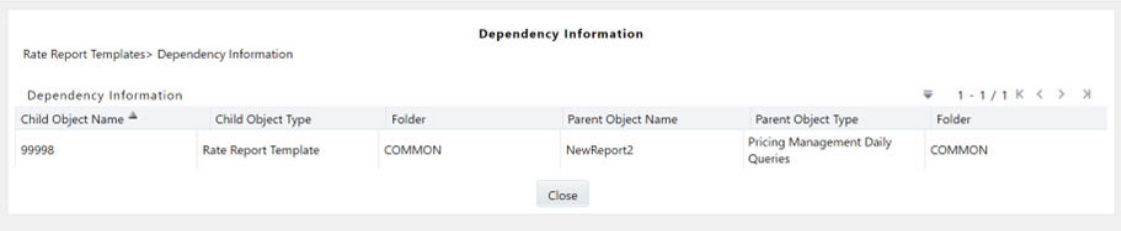
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-23 Deleting an Existing Rate Report Template



If template is currently is used in any of the rate card reports, it displays in following dependency information:

Figure 33-24 Deleting an Existing Rate Report Template – Dependency Information



33.4 Defining a Rate Report

The Rate Card window allows users to select the set of standard products, report templates and the relevant historical date range to include in their report.

The Rate Card contains a Rate Report definition page and a Report page. The Rate Report definition includes the name of the Rate report, the historical time horizon for reporting, report templates and the set of standard products for which the user wants to fetch the rates.

Each report can contains different set of columns based on selected report template. Columns could include the calculated base Transfer Rate and remaining term Transfer Rate, as well as relevant product characteristics defined for the respective standard products.

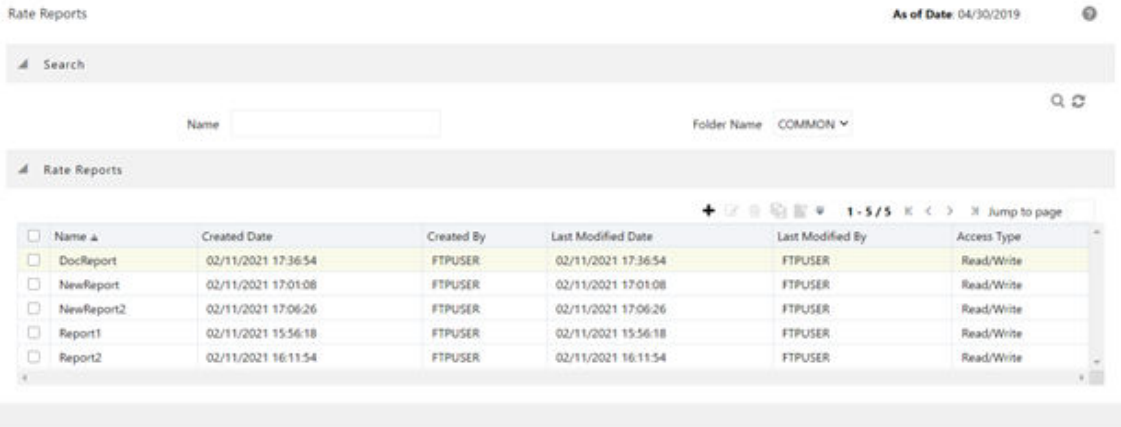
The procedure for working with and managing Rate Report is similar to that of other Oracle Funds Transfer Pricing business rules. It includes the following steps:

- Searching for Rate Reports
- Creating a Rate Report
- Viewing and Editing Rate Report
- Copying Rate Report
- Deleting Rate Report

Copyright © 2011, 2026, Oracle and/or its affiliates.

As part of creating and editing a Daily Rate Report, the user defines the products and historical date range.

Figure 33-25 Rate Reports



- Defining Rate Report
  To create a new Rate card:
- Defining a Rate Report: An Example
  The Rate Card window allows users to select the set of standard products, report templates and the relevant historical date range to include in their report.

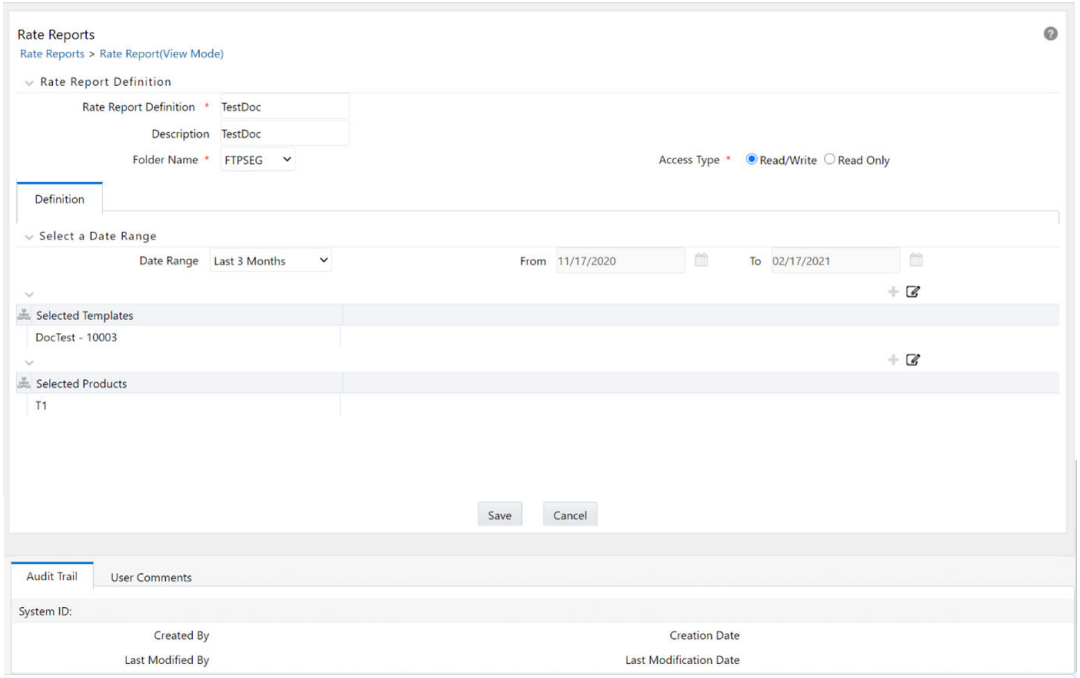
33.4.1 Defining Rate Report

To create a new Rate card:

1. Navigate to **FTP Rate Card** and select **Rate Report**.
2. Click the **Add** icon.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-26 Rate Report – Definition Mode



Prerequisites: Performing basic steps for creating or editing a Rate Card.

Procedure: This table describes the key terms used for this procedure.

Table 33-3 Fields and Descriptions from the Rate Report page

Term	Definition
Rate Report Definition	Enter the Rate Report name.
Description	Enter the description of the Rate Report.
Folder	Select the Folder name.
Access Type	Select the Access type of Rate Report as Read/ Write or Read Only.
Date Range	Select the date range for which you want to view the daily FTP rates. Here, you can give the specific date range also. The FROM and TO options will be enabled once you select the Specific Date Range option from the Date Range drop-down list.
Template Selection	Click template selection button to select already defined report templates to be included in your pricing report.
Product Selection	Click the Product Mapping (Add) icon to select the relevant standard products to be included in your pricing report. You can select one or more templates. However, at once, you can select only one product. If you want to add more, click the Edit icon and select another product.

3. Click **Save**.

Copyright © 2011, 2026, Oracle and/or its affiliates.

33.4.2 Defining a Rate Report: An Example

The Rate Card window allows users to select the set of standard products, report templates and the relevant historical date range to include in their report.

The Rate Card contains a Rate Report definition page and a Report page. The Rate report definition includes the name of the Rate Report, the historical time horizon for reporting, report templates and the set of standard products for which the user wants to fetch the rates.

Each report can contains different set of columns based on selected report template, columns could include the calculated base Transfer Rate and remaining term Transfer Rate, as well as relevant product characteristics defined for the respective standard products.

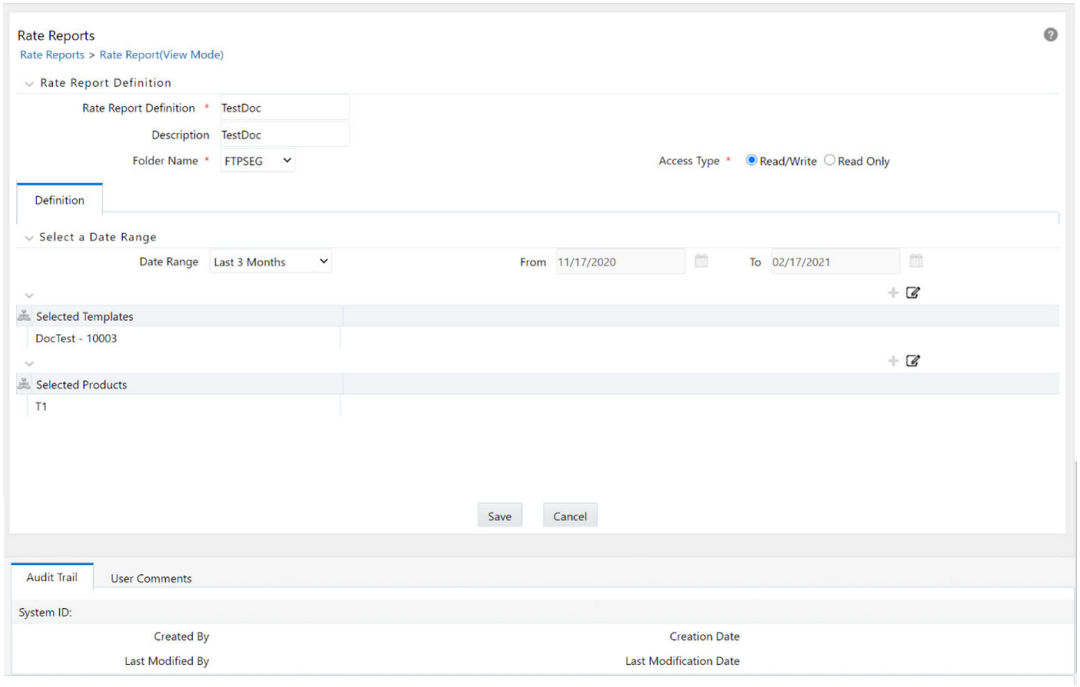
The procedure for working with and managing Rate Report is similar to that of other Oracle Funds Transfer Pricing business rules. It includes the following steps:

- Searching for Rate Reports
- Creating Rate Report
- Viewing and Editing Rate Report
- Copying Rate Report
- Deleting Rate Report

To define a Rate Report, follow these steps:

1. Enter the **Rate Report** name and **Description**.
2. Select the **Folder Name** and **Access Type** details.

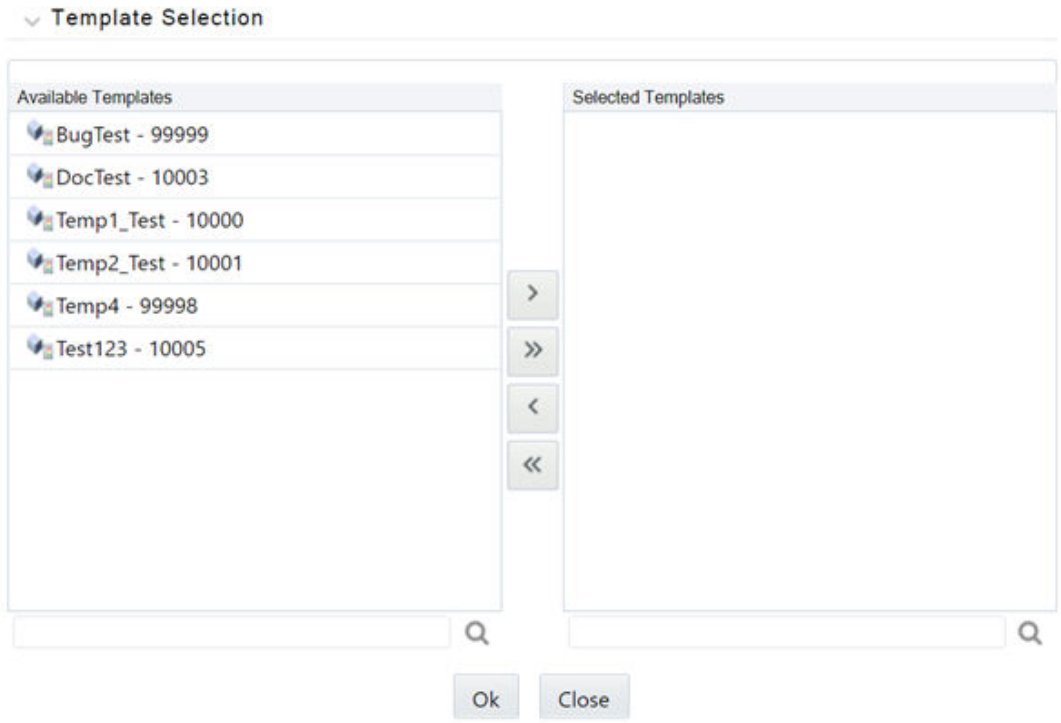
Figure 33-27 Rate Report – Definition Mode



3. Select the **Date Range**.
4. Select the report templates from template browser using the **Add** icon. This template browser will have the list of defined report templates.

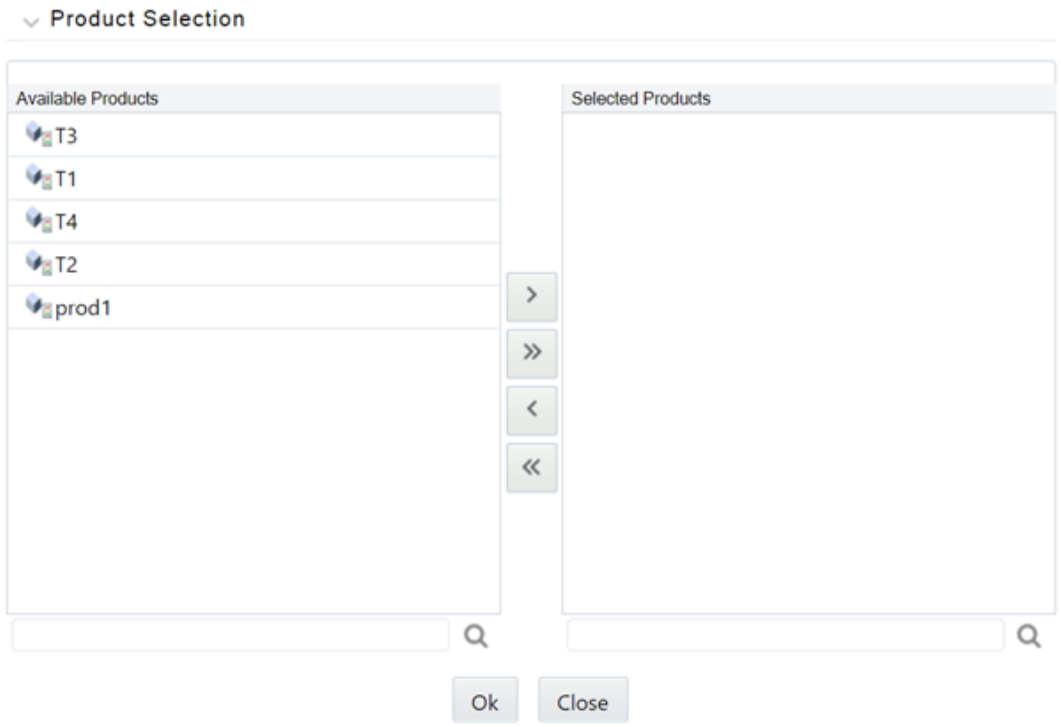
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 33-28 Template Selection – Hierarchy Browser



5. Select the **Product(s)** from the hierarchy browser using the **Add** icon. This hierarchy browser will have the list of defined standard products (as set up by the Administrator). This screen displays the template that have been created by different users. You can select any of the available templates.

Figure 33-29 Product Selection – Hierarchy Browser



Copyright © 2011, 2026, Oracle and/or its affiliates.

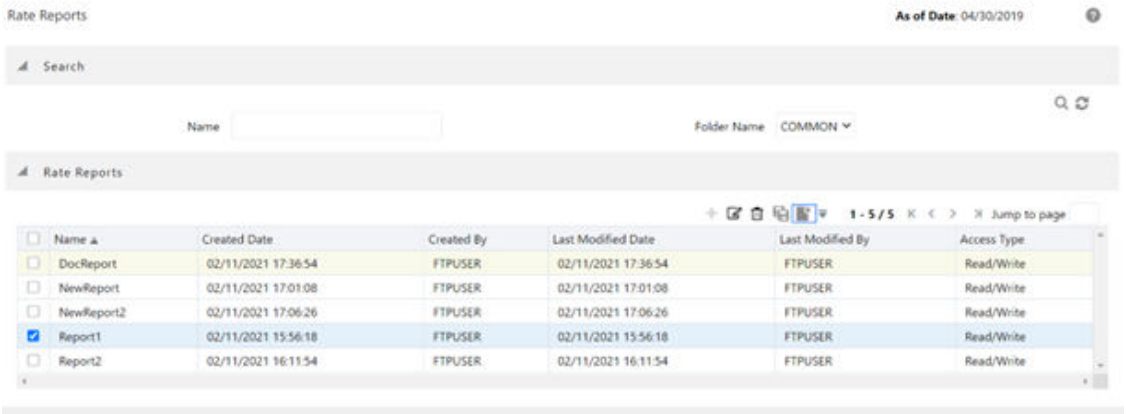
6. Click **Save**.
   You will be directed back to the Rate Report summary page and the defined Rate Report will be displayed on the Rate Report summary page.

33.5 Viewing Rate Card Report

Rate Card report is available upon successful completion of the Pricing Run.

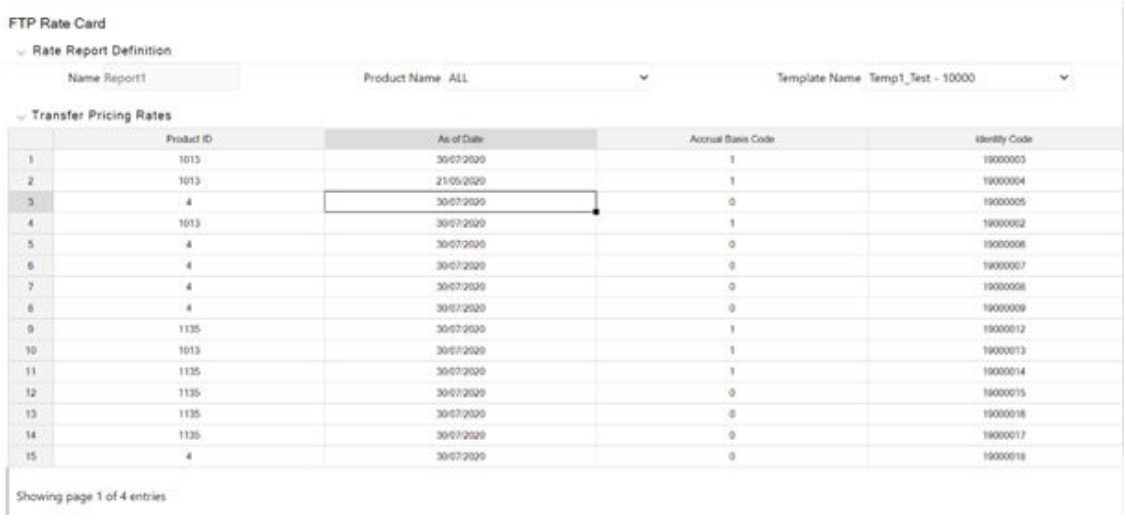
To access a Rate Card Report, select the Rate Report Name using the corresponding check-box and click the **View** icon.

Figure 33-30 Rate Reports



The Rate Card report is displayed.

Figure 33-31 Rate Card Report



Copyright © 2011, 2026, Oracle and/or its affiliates.

34

Data Aggregation and Stratification

Data Aggregation and Stratification enables you to summarize large volumes of financial instruments to a manageable scale for processing purposes. The engine that does the aggregation and stratification processing is the Data Stratification Rule. This application enables you to perform mathematically intensive calculations on a relatively small number of instrument pools that are proxies for a much larger number of individual instrument records. Using this feature is a multi-step process, which includes:

- Understanding what you want and how you want to aggregate and stratify data.
- Optionally defining balance or rate tiers to be used for grouping data.
- Configuring the Data Map metadata for the source and target data maps that are to be used on the Stratification Definition page.
- Setting up stratification definitions and rules in the Stratification Rule pages.
- Testing the rules that you create.

- Partial Aggregation
- Stratification Action Operations
- Aggregation Action Operations
- Stratification Tips
- Setting up Stratification Definitions

34.1 Partial Aggregation

An extra filter option is given in the Stratification Definition, which limits the stratification, only for the accounts in the Source Table falling under the filter criteria. If the Filter is not being used, (case for all existing definitions), then the existing behavior remain as is.

For accounts in the Source Table, which does not satisfy the filter condition will not be aggregated and will be moved to the destination table as is. But whether column mappings which are defined in aggregation tab and under defaults will be applicable or not for non-aggregated records, will be dependent on the check-box Apply Column mapping for Non-Aggregated Accounts. If this check-box is selected, then the aggregated records column mappings will be applicable to the non-aggregated records. Else the non-aggregated records will be moved to the destination aggregated table without using the column mappings

34.2 Stratification Action Operations

When creating pools of instruments, you need to identify the operation for each of the attributes. You have the following choices in terms of how to populate each attribute in the instrument pool from the instrument data:

- Discrete: Directs the Aggregation and Stratification engine to populate the instrument pool with discrete values.
  This option populates the pool with discrete values for an attribute if there is no logical or mathematical way to group instruments with different values, and the attribute is significant

Copyright © 2011, 2026, Oracle and/or its affiliates.

for processing purposes. Choosing the discrete action for an attribute ensures that only instruments with matching values are grouped in a pool. The discrete action may be the correct choice for important code fields. Be careful when using the discrete action on the Amount, Date, or Rate Fields. These types of fields tend to have many discrete values. A typical value for a discrete group by is a dimension ID field or a yes or no flag.

- Group by: Use for dimensions or other attributes that you want to use to group the detailed data.
  This option allows you to select columns that you want to use to group the instrument data. This will be used for dimension columns like Product, Org Unit or Currency, or any other dimensions that are needed for grouping the data.
- Tier: If you want to group the data by ranges of values, you can define the ranges using Stratification Tiers.
  Tiers are used for balance and rate fields to assist in grouping similar data into tranches or pools. Tiers should be defined on columns that affect the way assumptions are made, such as prepayments, interest rate margins, or other meaningful assumptions.

34.3 Aggregation Action Operations

The following are the default aggregation action operations:

- Average: Calculates the average value of an attribute for all instruments in the pool.
- Weighted Average: Calculates the average weighted value for an attribute by the current balance.

Examples of attributes for which you may want to calculate the weighted average are interest rate fields: interest rates, cap rates, floor rates, and interest rate margins for floating rates. Other possibilities include the remaining term, original term, loan to value, and so forth.

- Min, Max (minimum, maximum): Select one of these operators to populate the pool with the minimum or maximum value of a given attribute for all the instruments that are part of the pool.
- Median: Applies the Median Date value for all date values in the pool.
- **Sum** (The sum action calculates the total sum value for a numeric attribute.

The most common attributes that you want to calculate the sum for are the balance and payment amount fields. By calculating the total of the balance and payment amounts, the total of all pool balances should reconcile to and represent the totals of all instruments that are in the portfolio.

- Default: Applies a specified default value.

34.4 Stratification Tips

Consider this criteria when evaluating how you want to stratify data:

- When identifying attributes that you want to include as part of the instrument pools, keep in mind the primary use of the stratified data you are creating.
  For example, the ALM engine uses only the current balance amount fields for its calculations. As far as the ALM engine is concerned, designing a stratification rule that populates other balance or amount fields is optional.
- You need to identify the stratification criteria for the attributes.
  This step is simplified by the fact that any attributes that you identify as needing to maintain discrete values for, or that are going to be dropped or defaulted by definition, cannot be used as grouping criteria for pools.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Maximize efficient processing by designing rules to summarize the instruments into as few pools as possible, while at the same time only grouping instruments that generate the same or similar cash flows.

In other words, all instruments that are grouped into a pool should share common and pertinent financial characteristics.

> **Note**
>
> Unique Constraint on all Aggregated tables is the same as other instrument tables with ID Number and Identity Code. In order to preserve uniqueness, DEFAULT value should not be used for ID_NUMBER and IDENTITY_CODE mappings. MIN/MAX should be used for these fields accordingly.

34.5 Setting up Stratification Definitions

This section discusses how to setup stratification configurations.

To set up the Stratification Configurations, select **Common Object Maintenance**, select **Stratification**, and then select **Stratification Definition**. You can define the aggregation process at any level of complexity.

> **Note**
>
> When you to edit or Save As an existing stratification definition, all mapped columns will revert to un-mapped. To keep the mapping configurations intact, following file need to be edited:
>
> - Navigate to the deployment path as given the example.
> Ex: `scratch\FTP8074Test\tomcat\webapps\ofsa\conf\excludeURLList.cfg`
> - Configure `excludeURLList.cfg` file.
> - Add the statement below at the end of the file if it is already not present.
> `[SQLIA]./stratification/`
> - Save the `excludeURLList.cfg`.

- Stratification Configuration
- Setting up Stratification Definitions
- Defining Stratification Rules
- Defining Stratification Tiers
- Executing a Stratification Rule
- Creating New Aggregate Table

34.5.1 Stratification Configuration

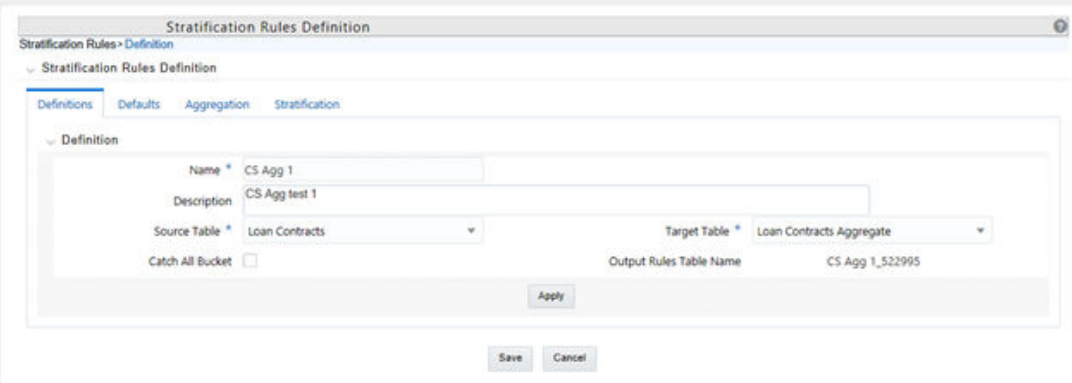
To define the aggregation process at any level of detail of complexity, follow these steps:

1. Navigate to **Common Object Maintenance**, select Stratification, select **Stratification Definition** and then select New or Edit.

Copyright © 2011, 2026, Oracle and/or its affiliates.

This example illustrates the fields and controls on the Stratification Definition page. You can find definitions for the fields and controls later in this section.

Figure 34-1 Stratification Rule – Definition page



2. Establish how you want the Stratification Engine to process amounts, rates, dates, and integers by assigning a stratification action to each.
   Default input applies a default value into the action settings on the Definition page.
3. Select a source and target instrument table, Default Values, Aggregation Actions, and Stratification Actions to complete the definition for each instrument table that you want to aggregate.

- Setting up Stratification Tier Structures

34.5.1.1 Setting up Stratification Tier Structures

An alternative to incremental grouping is to group according to tiers. You can use tiers on numbers, amounts, and rate fields. The main difference between using tiers versus other aggregation actions is that you can define the specific tiers that you want to stratify the data into to match business logic, reporting requirements, or both. Tiers also give you more control in terms of limiting the number of strata created for any given attribute to only ranges that are pertinent for processing or reporting purposes. For example, you could define a tier rule that groups the initial balance amounts in increments of 10,000. However, this could result in hundreds of strata if the balance amounts range anywhere from 1 to 100,000,000. A more efficient and logical way to group balance amounts may be to set up tiers so that the first tier contains records with amounts between 1 and 50,000; the second tier contains records with amounts between 50,000 and 1,000,000, and the third tier contains records with amounts between 1,000,000 and 100,000,000. Tiers are ideal for handling balance amounts.

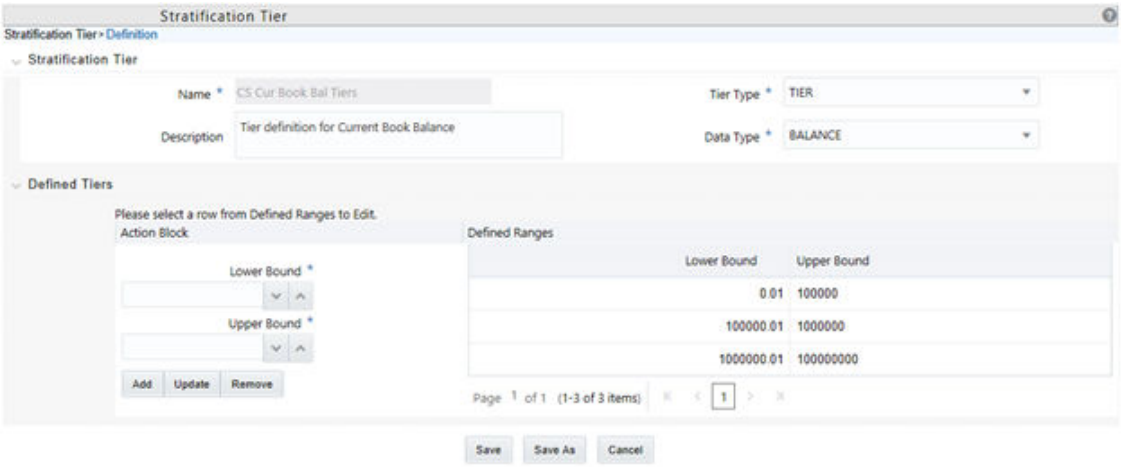
To set up Tier Structure, follow these steps, navigate to **Common Object Maintenance**, select **Stratification**, and then select **Stratification Tier**.

Stratify source data into a finite number of pools that are used by the Stratification Definition under Stratification Action.

Use the Tier Definition page to define stratification criteria for creating a finite number of pools that are used by the Stratification Definition - Group by page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 34-2 Stratification Tier – Definition page



This example illustrates the fields and controls on the Tier Structures page. You can find definitions for the fields and controls later in this guide.

Input the lower and upper boundary amounts for each tier and then add to the definition. The lower bound range must be greater than the prior upper bound amount. These values cannot be equal and do not overlap. The initial lower bound and last upper bound should small enough and large enough to capture all possible values.

Note that under **Tier Type**, Discrete is also available to select. This option populates the pool with discrete values for an attribute if there is no logical or mathematical way to group instruments with different values, and the attribute is significant for processing purposes. Choosing the discrete action for an attribute ensures that only instruments with matching values are grouped in a pool. The discrete action may be the correct choice for important code fields. Be careful when using the discrete action on the amount, date, or rate fields. These types of fields tend to have many discrete values. A typical value for a discrete group by is a dimension ID field, code field, or a yes/no flag.

34.5.2 Setting up Stratification Definitions

This section discusses how to setup stratification definitions.

Table 34-1 Pages Used to Setup Stratification Rules

Page Name	Navigation	Usage
Stratification Rules Definition - Definition	Common Object Maintenance > Data Stratification > Stratification Definition> Definitions	Specify rule name, description, source data, and target data tables.
Stratification Rules Definition - Defaults	Common Object Maintenance > Data Stratification > Stratification Definition> Defaults	Specify default values for each column type.
Stratification Rules Definition - Aggregation	Common Object Maintenance > Data Stratification > Stratification Definition> Aggregation	Define aggregation actions (or use the default values) for each column in the data map.
Stratification Rules Definition - Stratification	Common Object Maintenance > Data Stratification > Stratification Definition> Stratification	Define how to group or pool the data.

Copyright © 2011, 2026, Oracle and/or its affiliates.

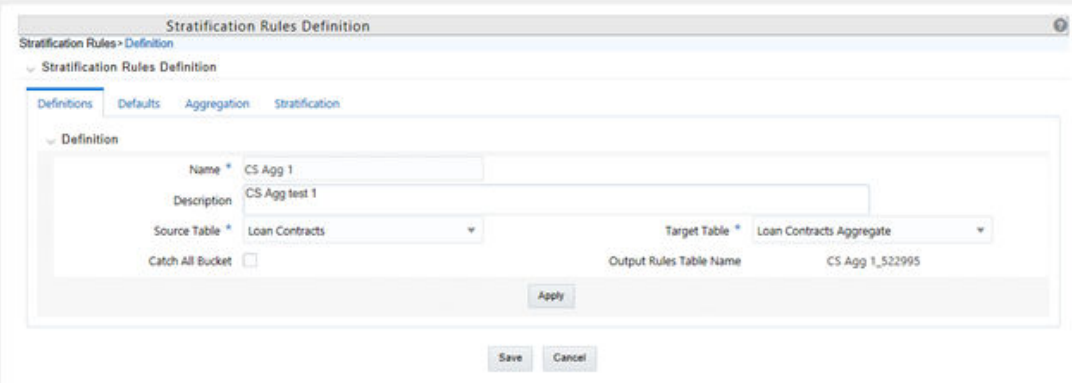
34.5.3 Defining Stratification Rules

Use the Stratification Rules - Tabs to define Stratification Definition.

To define Stratification Definition, follow these steps:

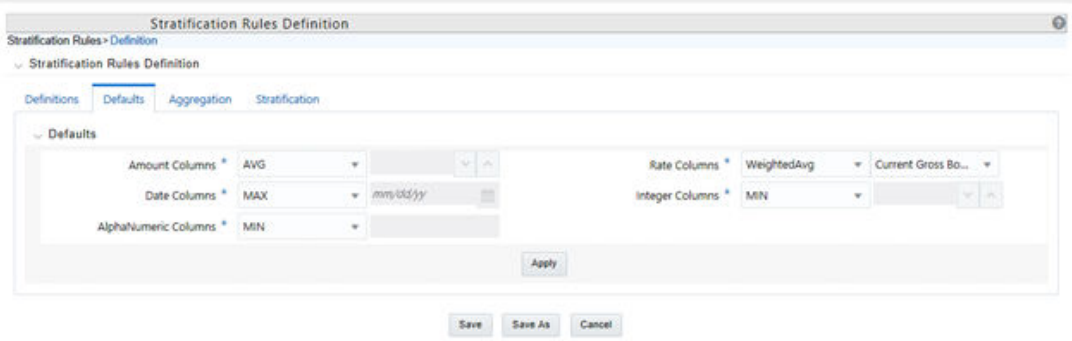
1. Navigate to **Common Object Maintenance**, select **Stratification**, and then select **Stratification Definition**.
   This example illustrates the fields and controls on the Stratification Rules - Definition page. You can find definitions for the fields and controls later on this page.

Figure 34-3 Stratification Rules Definition page



2. Enter a unique Name and Description for the Stratification Definition rule.
3. Select a Source Instrument Table. The list of values will include all seeded and properly registered instrument tables that contain account-level data records.
4. Select a Target Aggregate Instrument Table. This table will be populated with the results of the Stratification.
5. Specify whether or not to include a catch-all bucket in the process. This will ensure that all records from the source table are accounted for in the target table.
6. Click **Apply** to save your inputs and navigate to the next tab.
   This example illustrates the fields and controls on the Stratification Rules - Defaults Page.

Figure 34-4 Stratification Rules Definition – Defaults tab

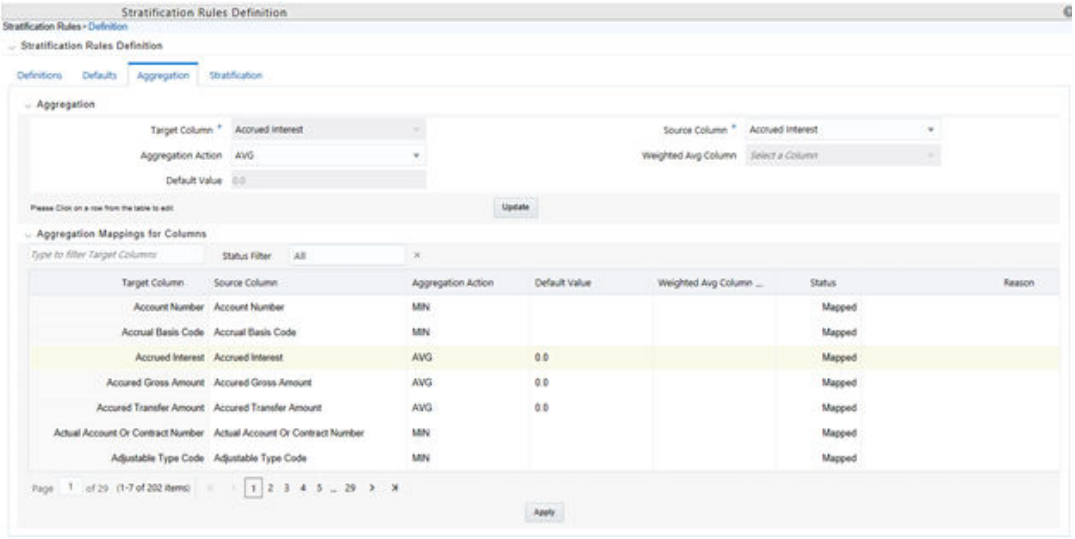


To define Stratification Defaults, follow these steps:

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. Select the Amount default. Possible selections are MAX, MIN, SUM, AVG. The recommended selection is SUM.
2. Select the Date default. Possible selections are MAX, MIN, MEDIAN. The recommended selection is MEDIAN.
3. Select the AlphaNumeric default. Possible selections are MAX, MIN. The recommended selection is MAX.
4. Select the Rate default. Possible selections are MAX, MIN, SUM, AVG, and Weighted Average. Recommended selection is Weighted Average. If Weighted Average is selected, then additionally select the balance column to use for weighting.
5. Select the Integer default. Possible selections are MAX, MIN. The recommended selection is MAX.
6. Select Apply to save your inputs and navigate to the next tab.
This example illustrates the fields and controls on the Stratification Rules - Aggregation Page.

Figure 34-5 Stratification Rules Definition – Aggregation tab



To define Stratification Aggregation details, follow these steps:

1. Select the specific row you want to define from the bottom half of the page.
You have the option to use the default values previously input on the Defaults tab or you can input a different Aggregation Action at the top of the page.
2. Use the pagination selections at the bottom of the page to address/review all columns or type the column name or a portion of the column name to navigate directly to a specific column.
3. Select Apply to save your inputs and navigate to the next page.
This example illustrates the fields and controls on the Stratification Rules - Stratification Page.

To define Stratification/Group by details, follow these steps:

1. Select the Column(s) and Stratification Action that you want to use for grouping the detailed records into aggregate/pooled records. Typical columns used for grouping will be dimension columns, balance or rate columns, or any other meaningful columns that you want to be unique for each pool record.

Copyright © 2011, 2026, Oracle and/or its affiliates.

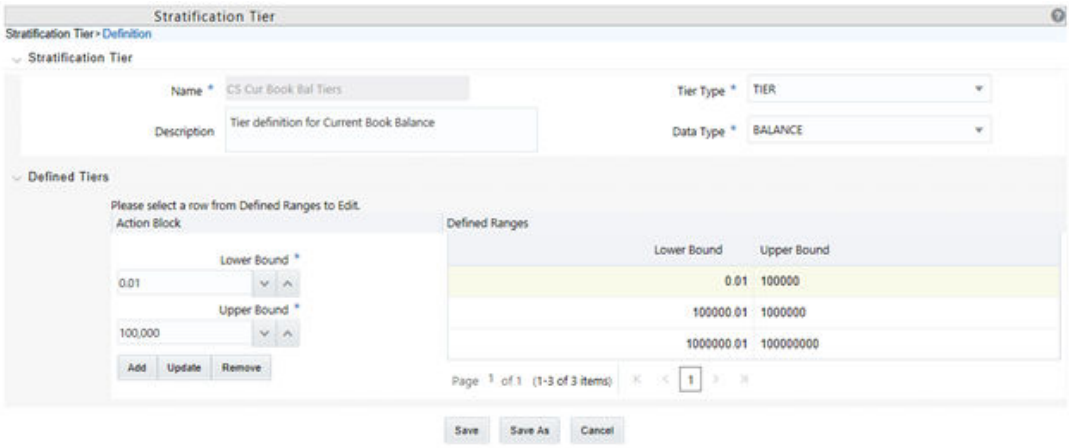
2. For each Stratification column, select the Stratification Action. Options are Tier, Discrete, and Group by. Both the Tier and Discrete options require an additional Stratification Tier definition to be selected.
3. Select Apply to save your inputs and remain in the rule or select Save to save your inputs and close the rule.

34.5.4 Defining Stratification Tiers

To define stratification tiers, follow these steps:

1. Navigate to Common Object Maintenance, select Stratification, and then select Stratification Tier.

Figure 34-6 Stratification Tiers - Fields and Controls



2. Provide a unique **Name** and **Description** for the Stratification Tier.
3. Select the **Tier Type**, for example, Tier or Discrete.
4. Select the **Column Data Type** that this definition will apply to. This selection will limit the columns where this Tier rule will be available for selection in the Stratification Definition Rule.
5. Define each tier with lower and upper bounds and add as a defined range. Click **Add** to add the tier. Additionally use Update or Remove to manage the individual tier definitions.
6. After all Tier ranges are defined, select Save to save the assumptions and exit the Tier screen.
   The tier definitions created in this step are mapped to the individual columns in the Stratification Definition rule on the Stratification tab.

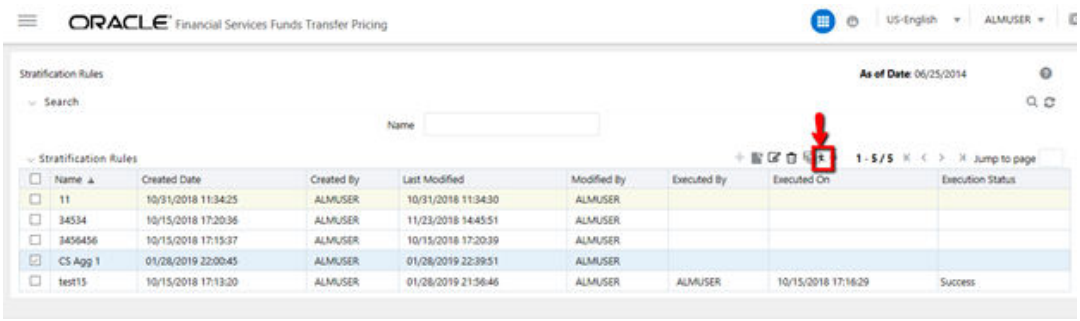
34.5.5 Executing a Stratification Rule

To execute a stratification rule, follow these steps:

1. Navigate to the **Stratification Rule Summary** page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 34-7 Stratification Rules



2. Select the Stratification Rule you want to process.
3. Select the **Run** icon.
4. Select the **As-of-Date** for the instrument data you want to aggregate.
5. Select **OK** to begin the process.

- Using SQL* Plus to Execute a Stratification Rule
- Using ICC Batch Framework

34.5.5.1 Using SQL* Plus to Execute a Stratification Rule

You can also execute the stratification rule either from SQL*Plus or from within a PL/SQL block OR from ICC Batch UI within the OFSAAI framework.

To Run the procedure from SQL*Plus, login to SQL*Plus as the Schema Owner. The function requires four parameters. Following is the syntax for calling the procedure:

```
declare
result number;
begin
---- Call the function
result := fn_insert_into_data_strat_tab(batch_id => :batch_id,
mis_date => :mis_date,
stratification_id => :stratification_id,
user_name => :user_name);
end;
```

- batch_id: any string to identify the executed batch
- mis_date: in the format YYYYMMDD
- stratification_id: numeric system id of stratification rule
- user_name: OFSAA user id of the user executing the function

Example:

```
declare
result number;
begin
---- Call the function
result := fn_insert_into_data_strat_tab(batch_id => 'abc',
mis_date => '20180601',
stratification_id => 123,
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

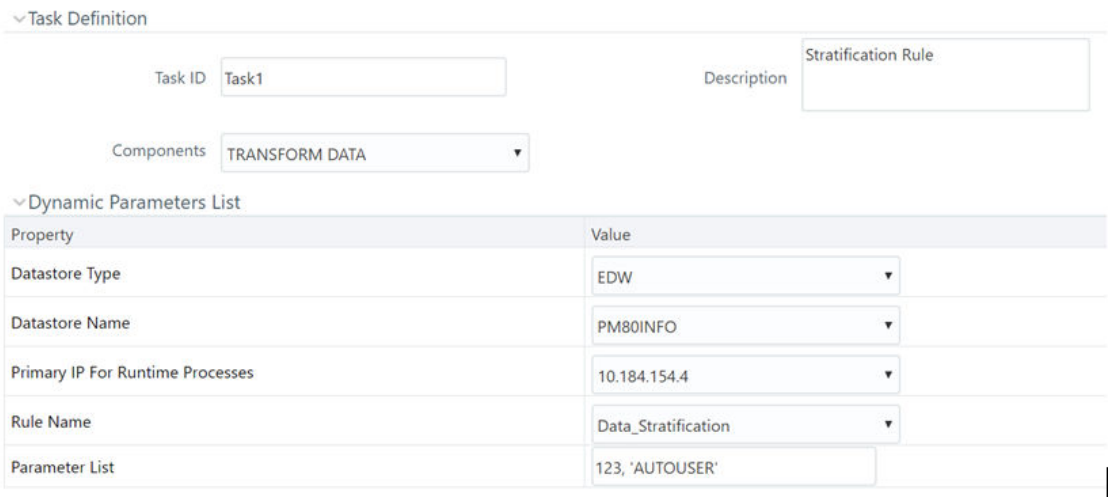
```
user_name => AUTOUSER);
end;
```

34.5.5.2 Using ICC Batch Framework

To execute the procedure from OFSAAI ICC framework, create a new Batch with task as TRANSFORM DATA and specify the following parameters for the task:

- **Datastore Type:** Select appropriate Datastore from the list
- **Datastore Name:** Select appropriate name from the list
- **IP Address:** Select the IP address from the list
- **Rule Name:** Data_Stratification
- **Parameter List:** stratification_id, user_name

Figure 34-8 Stratification using ICC Batch Framework



After the Stratification rule execution is complete, you can compare balances from the source table and target table to ensure all data has been accounted for, such as the sum of current balance from the source table should equal the sum of current balance in the target table.

After the data has been populated and verified in the aggregation tables, it is ready for processing by the ALM and (or) FTP applications. You can select either regular instrument tables or aggregate instrument tables in the ALM and TP Processes under Source Selection. Select only one. Selecting both in the same process will lead to double counting of the balances.

34.5.6 Creating New Aggregate Table

The following aggregate tables are seeded with the Application Data Model:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table 34-2 Seeded Aggregation Tables with the Application Data Model

#	Logical name	Physical table name
1	Annuity Contracts Aggregate	FSI_D_ANNUITY_CONTRACTS_AGGR
2	Borrowings Aggregate	FSI_D_BORROWINGS_AGGR
3	Checking And Savings Accounts Aggregate	FSI_D_CASA_AGGR
4	Credit Cards Aggregate	FSI_D_CREDIT_CARDS_AGGR
5	Credit Lines Aggregate	FSI_D_CREDIT_LINES_AGGR
6	Guarantees Aggregate	FSI_D_GUARANTEES_AGGR
7	Investments Aggregate	FSI_D_INVESTMENTS_AGGR
8	Leases Aggregate	FSI_D_LEASES_AGGR
9	Loan Contracts Aggregate	FSI_D_LOAN_CONTRACTS_AGGR
10	Money Market Contracts Aggregate	FSI_D_MM_CONTRACTS_AGGR
11	Mortgages Aggregate	FSI_D_MORTGAGES_AGGR
12	Retirement Accounts Aggregate	FSI_D_RETIREMENT_ACCOUNTS_AGGR
13	Term Deposits Aggregate	FSI_D_TERM_DEPOSITS_AGGR

To create a new aggregate table in addition to the basic instrument and leaf columns, the following mappings must be done in the data model:

- Super-type
- INSTRUMENT_AGGREGATE
- INSTRUMENT_AGGREGATE_BASIC
- Table classification
- EPM Aggregation (701)

This may be required when you create a custom instrument table and want to use stratification for it.

Copyright © 2011, 2026, Oracle and/or its affiliates.

35
Account Audit Portal

The Account Audit portal provides users a tool to validate account attributes along with calculated FTP results and the related cash flows:

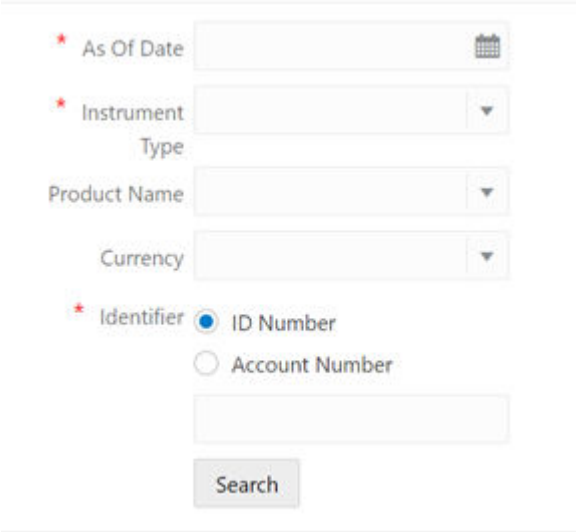
- Transfer Rates, Adjustment Rates, and Economic Cost results.
- Audit information like method used for transfer rate or adjustment rate calculations.
- Interest rate curves used for calculations.
- Cash flows generated for the account, which are helpful to verify the rate calculations.
- Process Error messages
- Current and Prior “As-of-Date” data is also accessible.

All these details can be verified in a single UI portal by functional users, without taking any technical help to query multiple DB tables.

To open the Account Audit

1. From the LHS menu, click **Funds Transfer Pricing**, and then select **FTP Account Audit**.

Figure 35-1 FTP Account Audit page



2. The following table describes the fields from the above screen.

Field	Description
As Of Date	By Default, this is the as of date set in application preferences. However, if you want to pull records for another date, you can choose from the calendar option.
Instrument Type	This is a mandatory field. By Default, this is blank. If you do not select any value from the drop-down list, you would not be able to proceed with your search.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Field	Description
	As, you can have 30 to 35 instrument tables to retain various instruments like separate tables for deposits, annuities, loans, saving accounts, and so on. Each table may have millions of records, so to speed up the process, you must select an instrument name from drop-down list, and corresponding table will be picked up for search.
Product Name and Currency	Both of these are optional fields with a drop-down. You can select a value to narrow-down the search criteria and reduce the turnaround time.
Identifier	You can enter either the ID Number or the Account number. You can select a particular radio button and enter the corresponding data.

3. After entering the relevant details, click **Search**.

   If there are no records for the search criteria entered, then FTP displays a message No records found. Update the search criteria.

   If the record is found, then the screen displays either the Account Number or ID Number based on your search criteria.

4. Select the **Account Number**. This enables an **Apply** button.
5. Click **Apply**. The system displays the selected Account Details.

   The Account Audit Details are displayed in four tabs:

   - Account Attributes
   - Audit Details
   - Cash Flows
   - Process Errors

- Account Attributes Tab
  The account attributes tab displays account/deal attributes, so you can get an idea if the account is fixed or floating rate, rate of interest charged, original balance, remaining balance, account maturity etc., which would be required to verify the calculations in subsequent tabs.
- Audit Details Tab
  The audit details tab contains two sections. The first section will display all the calculated rates, for example:
- Cash Flows Tab
  The Cash Flow tab details all the cash flows for the searched account. Same account can have multiple processes one for transfer rate, a different process for adjustment rate calculations, and so on, all processes which have generated cash flows for the record are listed here.
- Process Errors Tab
  The Process errors tab lists all the errors logged for the concerned account (Transfer Rate, Adjustment Rate, and Economic Cost Calculations, and so on). You could have Run the multiple processes for Transfer Rate or Adjustment Rate Calculations for the same account and the Process errors tab will collate all the errors spread across various processes for a particular account and display on one screen.

Copyright © 2011, 2026, Oracle and/or its affiliates.

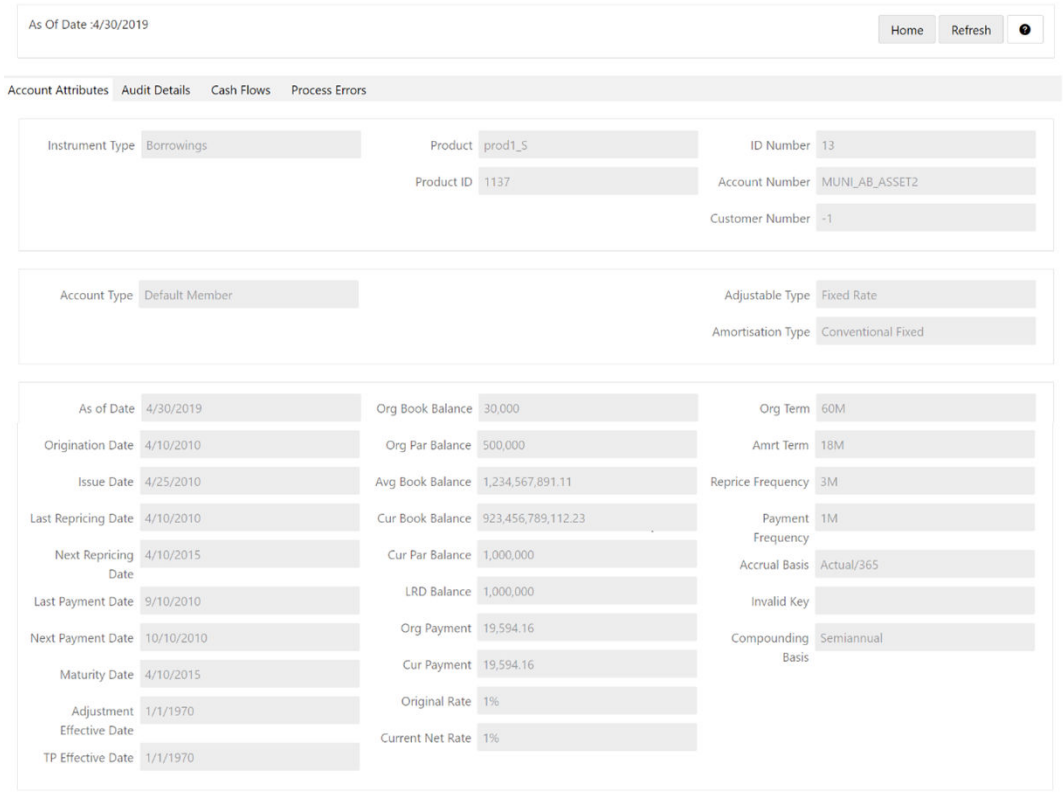
35.1 Account Attributes Tab

The account attributes tab displays account/deal attributes, so you can get an idea if the account is fixed or floating rate, rate of interest charged, original balance, remaining balance, account maturity etc., which would be required to verify the calculations in subsequent tabs.

This tab contains following details:

- Instrument Type
- Product Details
- Customer Details (ID Number, Account Number, and Customer Number)
- Account Type, Adjustable Type, Amortization Type
- Dates (As of Date, Origination Date, Issue Date, Last Reporting Date
- Balances, Payments, Terms, and Frequencies
- Customer Rate

Figure 35-2 Account Attributes Tab



35.2 Audit Details Tab

The audit details tab contains two sections. The first section will display all the calculated rates, for example:

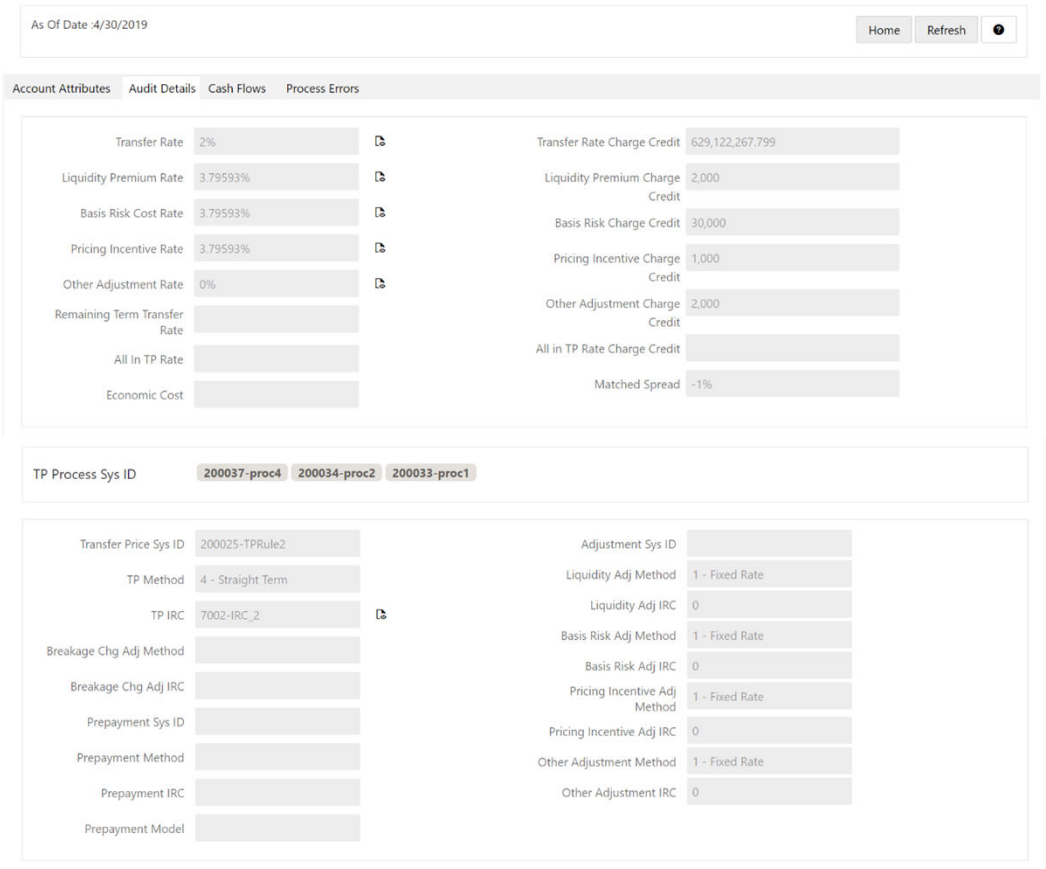
Copyright © 2011, 2026, Oracle and/or its affiliates.

- Transfer Rate
- Adjustments if any (Basis Risk Cost, Liquidity Premium, Other Adjustments or pricing incentive).
- Economic cost, in case any economic cost calculations are done.
- Corresponding charge credit details would also be displayed (if applicable).

Each rate is given with a view button next to it, in case you want to verify what method, rule, IRC, and so on is being used for particular rate calculations. When you click on the view button the corresponding details will be highlighted in the following section.

For example, if you want to verify Transfer rate calculation, you can click on view button next to it and verify which TP method, IRC is being used or if there is any prepayment assumption attached to it.

Figure 35-3 Audit Details Tab



35.3 Cash Flows Tab

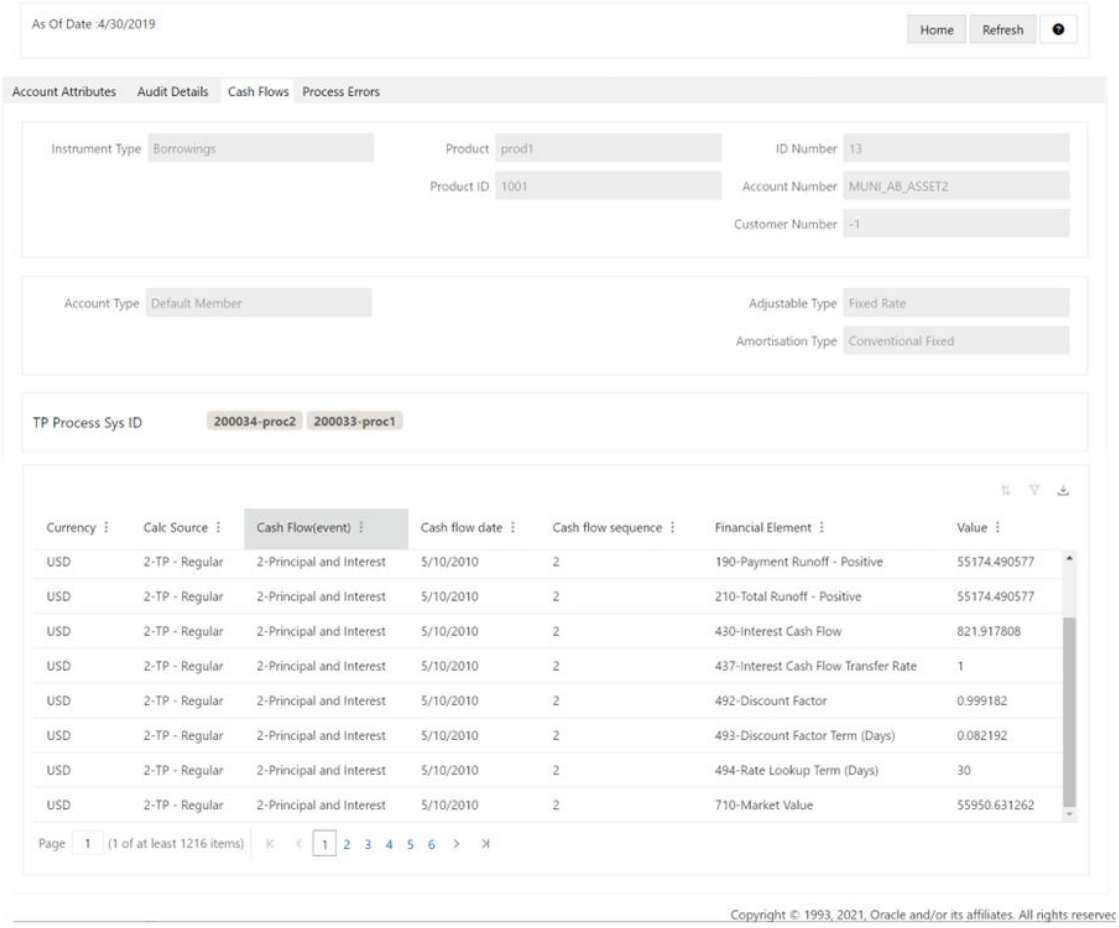
The Cash Flow tab details all the cash flows for the searched account. Same account can have multiple processes one for transfer rate, a different process for adjustment rate calculations, and so on, all processes which have generated cash flows for the record are listed here.

To identify the cash flows for a particular process, you can click on the process ID given next to “TP Process Sys ID” and cash flows only for selected process will be shown.

Copyright © 2011, 2026, Oracle and/or its affiliates.

You can download these cash flows in an excel, in case you want to revisit the calculations and confirm how cash flows are impacted due to prepayments or other details.

Figure 35-4 Cash Flow Details Tab



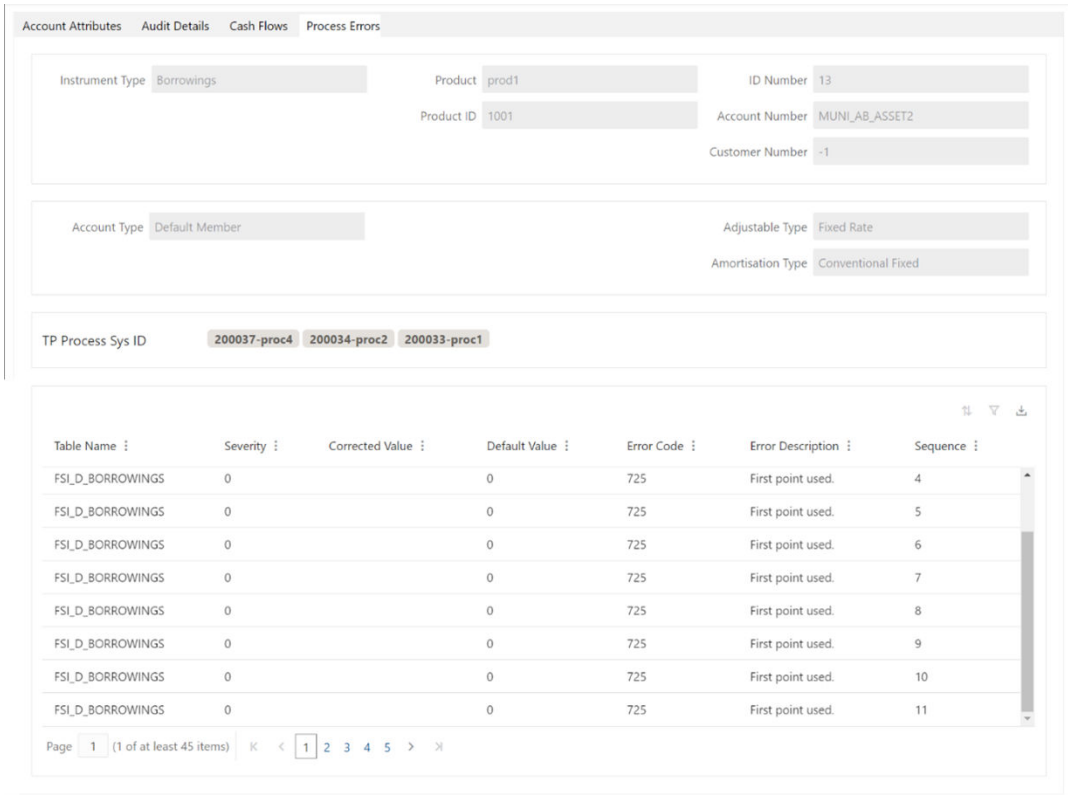
35.4 Process Errors Tab

The Process errors tab lists all the errors logged for the concerned account (Transfer Rate, Adjustment Rate, and Economic Cost Calculations, and so on). You could have Run the multiple processes for Transfer Rate or Adjustment Rate Calculations for the same account and the Process errors tab will collate all the errors spread across various processes for a particular account and display on one screen.

Similar to the cash flow tab, process errors also comes with process IDs' which can be selected to view errors corresponding to that particular process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 35-5 Process Errors Tab



The **Home** button takes you back to the landing page of FTP Account Audit homepage to search for another account.

The **Refresh** button refreshes the page in case there is some backend process running and simultaneously, updating the account records.

Copyright © 2011, 2026, Oracle and/or its affiliates.

36

View FTP Objects in Metadata Browser

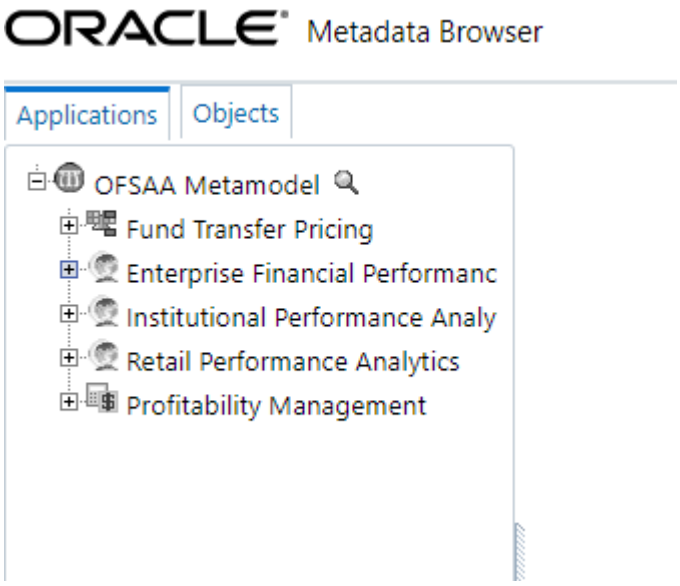
Metadata Browser (Object and Application View) provides a common repository of metadata Objects created in OFSAAI and applications hosted in OFSAAI. Using this view, you can identify the usage of base Objects in higher-level Objects and the mapping of Objects to Application. It enables to perform traceability and impact analysis. It also allows you to view the data flow and the workflow of the application and understand the usage of Objects within the application.

Metadata Browser (MDB) supports Application view and Object view: In the Application view, you can browse through the metadata created using the applications hosted in OFSAAI. In Object view, you can view the metadata created in OFSAAI.

To access the Metadata Browser (Object and Application View), your role Business Analyst (SYSBAU) must be mapped to the SCR_MDB function. For more information on mapping functions to a role, see Function - Role Map.

The Application view provides the top-down approach to view the defined metadata. The Application view shows the list of Applications at the parent level, and the Child elements are Runs for the application. This list will grow dynamically based on the Run definitions mapped to Application, through this tab users can view Run execution details in Metadata Browser.

Figure 36-1 Metadata Browser



For more information on Metadata Browser, see the Metadata Browser section in the OFS Analytical Applications Infrastructure User Guide.

- Configure Metadata Browser
  To configure the Metadata Browser for FTP

Copyright © 2011, 2026, Oracle and/or its affiliates.

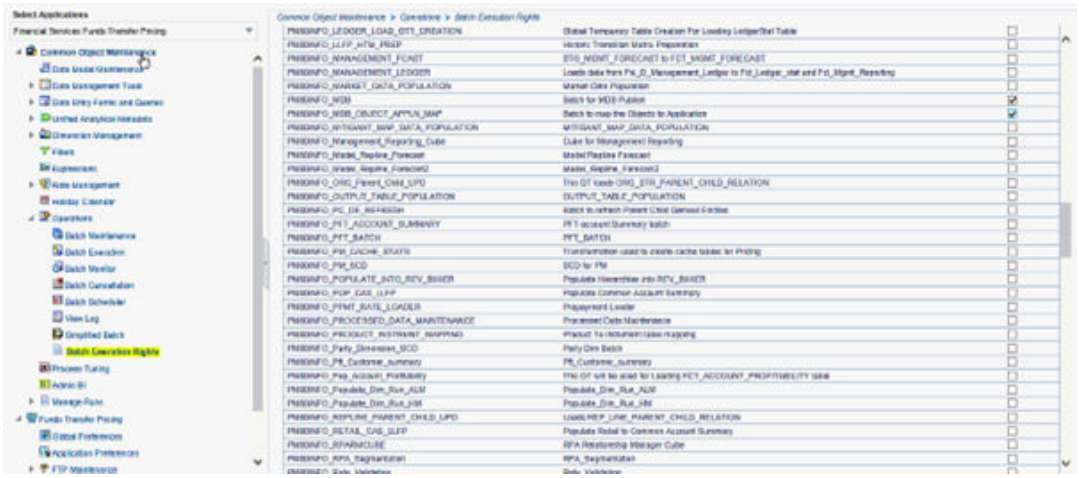
- Viewing FTP Objects in Metadata Browser
  Oracle Funds Transfer Pricing under Oracle Financial Services Analytical Applications has the Metadata Browser (MDB).

36.1 Configure Metadata Browser

To configure the Metadata Browser for FTP

1. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Execution Rights** to Map the MDB batch to User - Group.

Figure 36-2 Batch Execution Rights

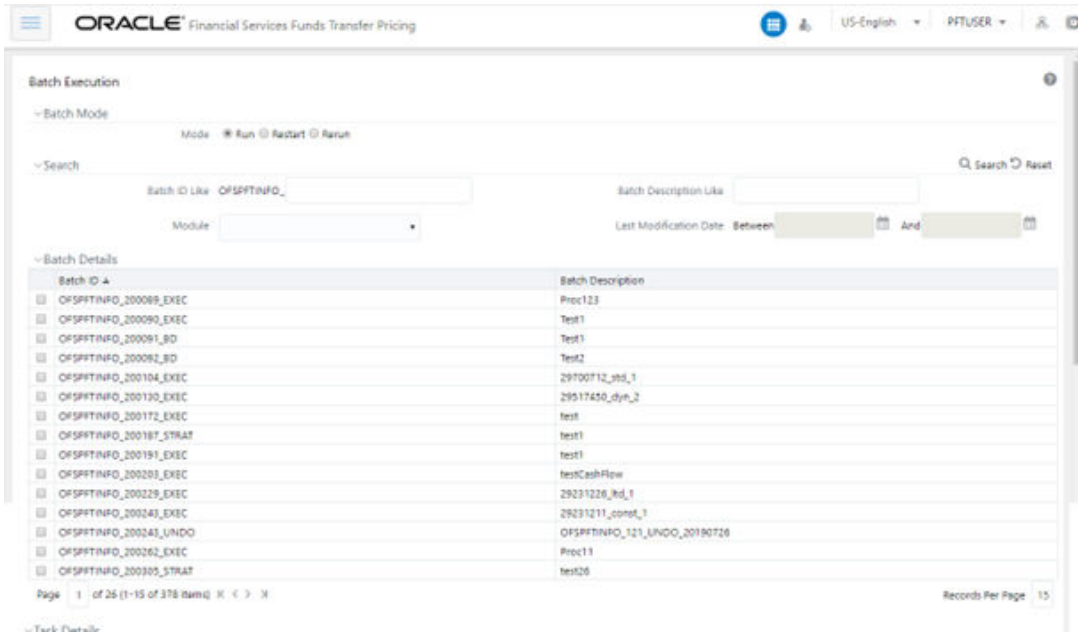


For more information, see *Command Line Utility to Publish Metadata in Metadata Browser* section in OFS Analytical Applications Infrastructure User Guide.

2. Execute the Batch for MDB publish.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 36-3 Batch Execution



For more information, see the *ICC Batch Execution* section in the OFS Analytical Applications Infrastructure User Guide.

3. Navigate to Metadata Browser (Object and Application) window of respective Application.

36.2 Viewing FTP Objects in Metadata Browser

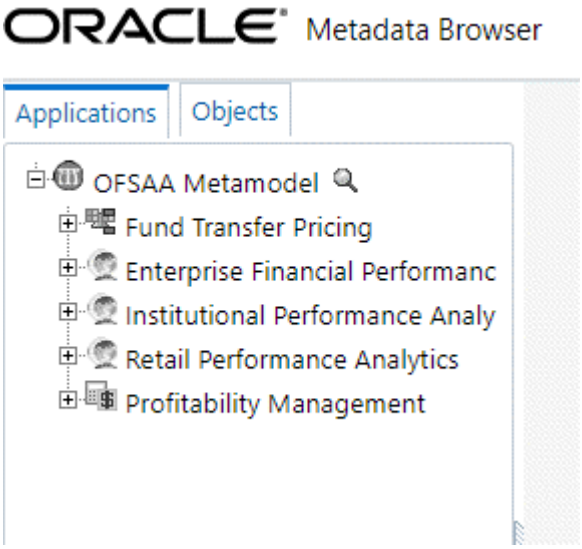
Oracle Funds Transfer Pricing under Oracle Financial Services Analytical Applications has the Metadata Browser (MDB).

To view FTP applications and objects in MDB, follow these steps:

1. After successful execution of the Batch for MDB Publish, select **Object Administration**, and then select **Metadata Browser**.
2. Click the **Application** tab under the Metadata Browser window to view the FTP application.

Copyright © 2011, 2026, Oracle and/or its affiliates.

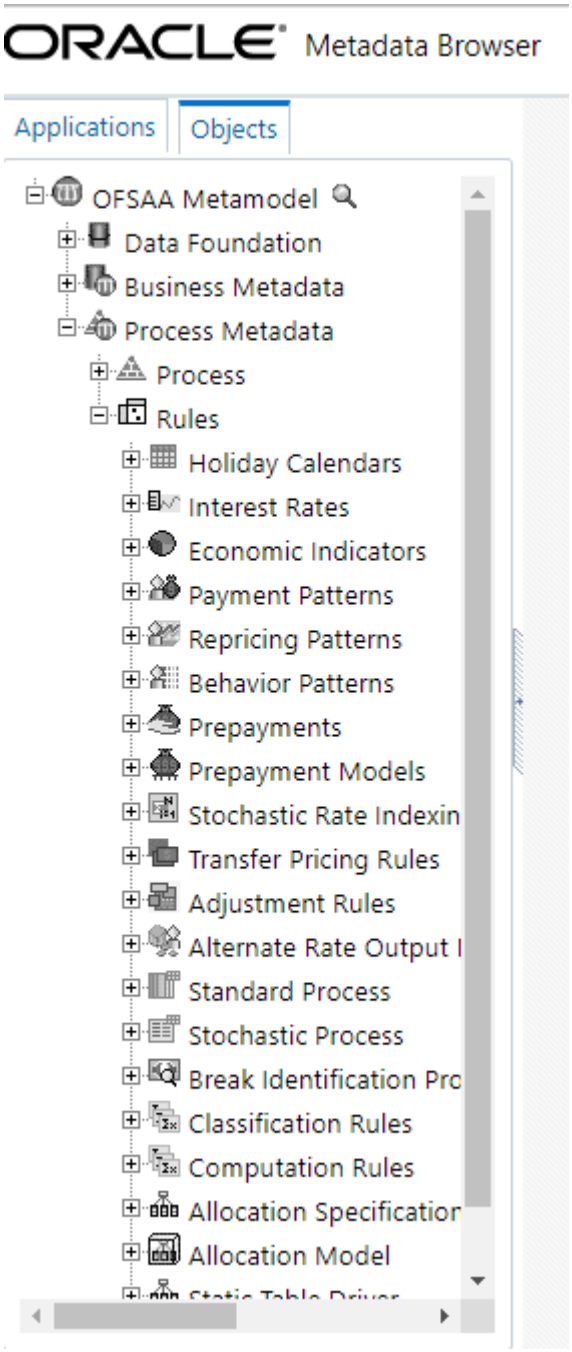
Figure 36-4 Metadata Browser – Applications tab



3. Click the **Object** tab under the Metadata Browser window to view FTP objects:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure 36-5 Metadata Browser – Objects tab



4. Under Process Metadata, and then under Rules, all the rules defined under the FTP window are displayed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

A
Management Ledger

This appendix covers the following topics:

- Statement of Direction
- Frequently Asked Questions
- Statement of Direction
  From Release 8.1, the Management Ledger replaces the Ledger/Stat table as the central repository for management accounting within OFSAA Enterprise Performance Management solutions.
- Frequently Asked Questions
  This topic covers the frequently asked questions about the Management Ledger.

A.1 Statement of Direction

From Release 8.1, the Management Ledger replaces the Ledger/Stat table as the central repository for management accounting within OFSAA Enterprise Performance Management solutions.

> **Note**
>
> Ledger/Stat will continue to be supported for existing or upgrading customers, but it will not be included as part of a new installation.
>
> There is a utility available for migrating the allocations defined in the Ledger/Stat table to the FSI_D_MANAGEMENT_LEDGER table. For more details, see the MOS Doc ID 2398698.1.

The new Management Ledger supports the same functionality as historically has been supported by ledger/stat and stores the same kinds of data as has always been housed in ledger/stat including: General Ledger Data:

- General Ledger Data:
  - Reconciled to instrument level data
  - Provides a starting point for management accounting processes
- Budget or Forecast Data:
  - From OFSAA Balance Sheet Planning or other budgeting and forecasting sources
- Multidimensional Statistical or Volumetric Data:
  - Used or created by allocation rules
- Value-Added data From:
  - OFSAA Profitability Management allocation rules
  - OFSAA Funds Transfer Pricing processes

Copyright © 2011, 2026, Oracle and/or its affiliates.

  - Other value-added processes

In the future, functional enhancements to Profitability Management will be directed towards the Management Ledger table class. New enhancements may or may not be extended to apply to the Management Ledger table.

A.2 Frequently Asked Questions

This topic covers the frequently asked questions about the Management Ledger.

How do OFSAA Infrastructure and Applications interact with the Management Ledger table?

- Loading Staging Data into the Management Ledger Table:
  - OFSAA Infrastructure 8.0.4.0.0 includes a T2T rule (T2T_MANAGEMENT_LEDGER) for loading the Management Ledger table from STG_GL_DATA
- OFSAA Profitability Management:
  - Distribution of balances within the management ledger level
  - Aggregation and posting of instrument or transaction summary-level data to the management ledger level
  - Distribution of ledger-level balances to the instrument or transaction summary level
  - Any other kind of Profitability Management process involving management ledger level data
- OFSAA Funds Transfer Pricing:
  - Funds Transfer Pricing supports the same functionality with the Management Ledger as historically has been available with the table
    * Direct transfer pricing of ledger-level balances
    * Aggregation/migration of instrument-level transfer pricing results to the ledger-level (weighted average rates and charges & credits for multiple FTP components)
- OFSAA Enterprise Financial Performance Analytics:
  - OFSAA Enterprise Financial Performance Analytics includes data movement processes for loading its results tables from either or Management Ledger

What are the functional advantages of the new table structure?

- Enhanced multi-currency model:
  - Functional & Entered balances for each row
  - More straightforward integration to multi-currency General Ledger systems
  - Simplified reconciliation of ledger-level data with multi-currency instrument data
  - All value-add processes post simultaneously to both Entered & Functional balances:
    * Funds Transfer Pricing
    * Profitability Management
- Daily posting/monthly processing: Supports daily posting to the Management Ledger level
  - Particularly useful for customers who want to post daily FTP results to the ledger level
- Multi-Entity/Multi-Tenant functionality: Multi-entity functionality, also introduced in release 8, is available with the Management Ledger table but is not supported.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Future Enhancements: All future enhancements to OFSAA applications that utilize ledger-level data will be built to support the Management Ledger structure.

What are the technical advantages of the new table structure?

The structure of the Management Ledger table eliminates the need to:

- Generate YTD balances for every row at the time of insertion.
- Refresh YTD balances for every row at the time of update.
- Store YTD balances on every row.

YTD balances may still be referenced in Profitability Management rules:

- The engine can easily generate this data when needed but the cost of calculation and storage for every row is eliminated.

Can I have multiple Management Ledger tables?

Yes. The Management Ledger table is seeded but Management Ledger is also a table class. If you do establish multiple Management Ledger tables, you must manage the data in each table and its relationship to instrument level data or other data in your Information Domain. For more information, see the *Adding Management Ledger Class Tables* section in the OFS Data Model Utilities User Guide.

Will customers need multiple Management Ledger tables?

Most customers will not need multiple Management Ledger tables. Utilization of multiple Management Ledger tables might be useful for some multi-entity or multi-tenant implementations.

Can different Management Ledger tables have different dimensions?

Yes, different Management Ledger tables may have different dimensionality.

Can I move data between and the Management Ledger table?

No. All rules and processes that post to or read from the ledger-level can utilize only one ledger table.

How long will the table be supported?

- Release 8 and all the previous OFSAA releases will always support the table and its related application functionality.
- And all the related application functionality may not be supported in future releases, but and its related application functionality will be supported for current releases and Oracle intends to continue supporting for the foreseeable future. New application enhancements are unlikely to be extended to the table.

Which table does Oracle recommend that customers use?

New Customers:

- Customers beginning new implementations are advised to use the new Management Ledger table and are advised not to use the table.
- Customers who elect to use should be aware that support for application functionality that utilizes the table would be withdrawn in a future release.
- Customers who want to migrate to a future release for which is no longer supported will be required to migrate their processes to the Management Ledger before upgrading to that release.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- The need to eventually move to the Management Ledger structure can be avoided by starting with the Management Ledger table.

Upgrading Customers:

- Customers upgrading or considering upgrading to release 8 from a prior release are advised to continue using the table.
- This will minimize the time and effort required for the migration to release 8 by retaining all rules and processes that currently involve the table.
- Customers upgrading to release 8 should be aware that support for application functionality that utilizes the table may be withdrawn in a future release.
- Application rules and processes that involve the table may need to be migrated from the Management Ledger table in the future.

Copyright © 2011, 2026, Oracle and/or its affiliates.

B
Multi-Language Support

With the introduction of Multi-Locale Support, user interfaces can be viewed in various languages. The display language is a selection on the login screen.

In addition to the base language (American English, for example, en_US), the following languages are currently supported for Funds Transfer Pricing, by installing the associated Language Pack.

Table B-1 Languages and Java Locales

Language Name	Java Locale
Arabic	AR_EG
Portuguese	PT_BR
French	FR_FR
German	DE_DE
Indonesian	IN_ID
Italian	IT_IT
Japanese	JA_JP
Korean	KO_KR
Russian	RU_RU
Simplified Chinese	ZH_CN
Spanish	ES_ES
Thai	TH_TH
Traditional Chinese	ZH_TW
Vietnamese	VI_VN
Turkish	TR_TR
Hebrew	iw_IL
French Canada	fr_CA

Depending on the type of the rule, the Name/Description specified during create/edit is either attached to the login-locale or is stored independent of the login-locale. The selection of the login-locale, therefore, governs which assumption/rules are visible, and which are not visible.

The following illustrates the visibility of assumptions and rules, across languages in the 8.0 release. This applies to all summary screens, embedded selections in subsequent screens, and so on.

By default, all Dimension Management Objects, as well as Filters and Expressions, would be visible in all languages (and therefore available for embedding), regardless of the language in which they were created.

Table B-2 Visibility of Assumptions and Rules

Assumption/Rules	Visible across all languages	Visible only in the creation-language*
Common Object Maintenance		

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table B-2 (Cont.) Visibility of Assumptions and Rules

Assumption/Rules	Visible across all languages	Visible only in the creation-language*
Process Tuning	Yes	
Simplified Batch		Yes
Common Object Maintenance> Rate Management		
Interest Rate Codes**	Yes	
Currencies**	Yes	
Funds Transfer Pricing		
FTP Maintenance		
Behavior Patterns	Yes	
Payment Patterns	Yes	
Repricing Patterns	Yes	
Propagation Pattern	Yes	
Assumption Specification		
Transfer Pricing Rules		Yes
Adjustment Rules		Yes
Prepayments		Yes
Prepayment Models		Yes
Alternate Rate Output Mapping		Yes
Stochastic Rate Indexing		Yes
FTP Processing		
Standard Process		Yes
Stochastic Process		Yes
Cash Flow Edits		Yes
Break Identification Process		Yes

Note

A future release (TBD) will expand visibility for many of these application object types, to provide visibility regardless of logged-in locale.
These object-types are visible across all languages because the Name and Descriptions are not stored in translation-compatible structures, which is different from the storage format for AAI objects like Dimension Management, Filters, and Expressions.

Copyright © 2011, 2026, Oracle and/or its affiliates.

C

Standard Navigation Paths

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

Oracle Financial Services Funds Transfer Pricing navigation paths are shown in this table. Access all of these pages through the Oracle Funds Transfer Pricing Administrator (FTP Administrator) or Oracle Funds Transfer Pricing Analyst (FTP Analyst) responsibility.

Table C-1

Page	Navigation Path
Application Preferences Home	FTP Maintenance > Application Preferences
Transfer Pricing Rule Home	Assumption Specification > Transfer Pricing Rules
Transfer Pricing Methodology	Assumption Specification > Transfer Pricing Rules > New > TP Method Selector
Cash Flow Edits Process	FTP Processing > Cash Flow Edits
Behavior Pattern Home	FTP Maintenance > Patterns > Behavior Patterns
Payment Pattern Home	FTP Maintenance > Patterns > Payment Patterns
Repricing Pattern Home	FTP Maintenance > Patterns > Repricing Patterns
Adjustment Rule Home	Assumption Specification > Adjustments
Adjustment Methodology	Assumption Specification > Adjustments > New > Adjustment Method Selector
Alternate Rate Output Mapping Rule Home	Assumption Specification > Alternate Rate Output Mapping
Interest Rate Code Home	Common Object Maintenance > Rate Management > Interest Rates
Monitor Requests	Process Management > Requests > Monitor
Stochastic Rate Index Rule Home	Assumption Specification > Stochastic Rate Indexing
Prepayment Rule Home	Assumption Specification > Prepayments
Prepayment Methodology	Assumption Specification > Prepayments > New > Prepayment Method Selector > Add New
Prepayment Models Home	Assumption Specification > Prepayment Models
Propagation Pattern Home	FTP Maintenance > Patterns > Propagation Pattern
Standard Transfer Pricing Process Home	FTP Processing > Standard Process
Stochastic Transfer Pricing Process Home	FTP Processing > Stochastic Process
Currency Rates Home	Common Object Maintenance > Currency Rates
Currencies Home	Common Object Maintenance > Currency

Copyright © 2011, 2026, Oracle and/or its affiliates.

D
Archive-Restore

This appendix covers the following topics:

- Command Line Migration
- Command Line Migration Utility
- Supported Objects
- Input Value for OBJECTMIGRATION.xml
- Logging
- Command Line Migration
  Using the command-line utility, you can migrate (export/ import) FTP metadata objects across different OFSAA environments. You can specify one or more objects within an object type or multiple object types.
- Command Line Migration Utility
  Oracle Funds Transfer Pricing uses the same utility that is used by the OFSAA infrastructure.
- Supported Objects
  The following objects are supported for migration:
- Input Values for OBJECTMIGRATION.xml
  The following table describes the input values required for the OBJECTMIGRATION.xml file:
- Logging

D.1 Command Line Migration

Using the command-line utility, you can migrate (export/ import) FTP metadata objects across different OFSAA environments. You can specify one or more objects within an object type or multiple object types.

> **Note**
>
> While importing any existing assumption rule, system regenerates the conditional assumption sys IDs. Even if user retries the import with the same dump file, these conditional assumptions are removed and re-inserted again with new conditional sys IDs.

D.2 Command Line Migration Utility

Oracle Funds Transfer Pricing uses the same utility that is used by the OFSAA infrastructure.

For more details, see the Command Line Utility to Migrate Objects section in the Object Migration chapter of the OFS Analytical Applications Infrastructure User Guide.

Copyright © 2011, 2026, Oracle and/or its affiliates.

D.3 Supported Objects

The following objects are supported for migration:

Table D-1 Supported Objects for Migration

Object Name	Object Type ID	Support for wild card Select all option	Support for implicit Dependency	Object Code	Remarks
Application Preferences	201	No	No	CONFIG	The entire data set is exported and replaced in the target. No validation and its dependency objects are handled.
Propagation	810	No	No	CONFIG	The entire data set is exported and replaced in the target. No validation and its dependency objects are handled.
Repricing Patterns	808	Yes	Yes	User-defined unique code	
Payment Patterns	809	Yes	Yes	User-defined unique code	
Behavior patterns	807	Yes	Yes	User-defined unique code	
Holiday Calendar	815	Yes	Yes	User-defined unique code	
Interest Rates	801	Yes	Yes	User-defined unique code	Historical rates from the FSI_IRC_RATE_HIST table are not migrated.
Currency rates	803	No	No		Currency rates are not supported.
Currencies	802	Yes	No	User-defined Currency code	
Economic Indicator	804	Yes	Yes	System generated code	FSI_ECONOMIC_COST_HIST is not migrated.
Transfer Pricing Rules	200	Yes	Yes (Except MAP objects)	System generated code	If Conditional Assumption uses MAP objects, then they should be explicitly migrated using Archive restore.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table D-1 (Cont.) Supported Objects for Migration

Object Name	Object Type ID	Support for wild card Select all option	Support for implicit Dependency	Object Code	Remarks
Adjustment Rules	401	Yes	Yes ((Except MAP objects)	System generated code	If Conditional Assumption uses MAP objects, then they should be explicitly migrated using Archive restore.
Prepayment Models	312	Yes	Yes	System generated code	
Prepayment Rules	300	Yes	Yes	System generated code	
Stochastic Rate Index Rules	315	Yes	Yes	System generated code	
Alternate Rate Output Mapping Rules	402	Yes	Yes	System generated code	
FTP Standard Processes	204	Yes	Yes	System generated code	
FTP Stochastic Processes	203	Yes	Yes	System generated code	
Break Identification Processes	816	Yes	Yes	System generated code	
Cash Flow Edits	700	Yes	Yes	System generated code	
FTP Rate Card Products	605	No	No	CONFIG	The entire data set is exported and replaced in the target. No validation and its dependency objects are handled.
Rate Reports	604	Yes	Yes	System generated code	

D.4 Input Values for OBJECTMIGRATION.xml

The following table describes the input values required for the OBJECTMIGRATION.xml file:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table D-2 Input Values for OBJECTMIGRATION.xml File

Object Name	Object Type Id	Input value in OBJECTMIGRATION.XML for the wild card	Input value in OBJECT_MIGRATION.xml for Object Code
Application Preferences	201	<OBJECT Code="CONFIG" Type="201"/>	<OBJECT Code="CONFIG" Type="201"/>
Propagation	810	<OBJECT Code="CONFIG" Type="810"/>	<OBJECT Code="CONFIG" Type="810"/>
Repricing Patterns	808	<OBJECT Code="*" Type="808"/>	<OBJECT Code="User defined unique code" Type="808"/>
Payment Patterns	809	<OBJECT Code="*" Type="809"/>	<OBJECT Code="User defined unique code" Type="809"/>
Behavior Patterns	807	<OBJECT Code="*" Type="807"/>	<OBJECT Code="User defined unique code" Type="807"/>
Holiday Calendar	815	<OBJECT Code="*" Type="815"/>	<OBJECT Code="User defined unique code" Type="815"/>
Interest Rates	801	<OBJECT Code="*" Type="801"/>	<OBJECT Code="User defined unique code" Type="801"/>
Currencies	802	<OBJECT Code="*" Type="802"/>	<OBJECT Code="User defined Currency code" Type="802"/>
Economic Indicators	804	<OBJECT Code="*" Type="804"/>	<OBJECT Code="System generated code" Type="804"/>
Transfer Pricing Rules	200	<OBJECT Code="*" Type="200"/>	<OBJECT Code="System generated code" Type="200"/>
Adjustment Rules	401	<OBJECT Code="*" Type="401"/>	<OBJECT Code="System generated code" Type="401"/>
Prepayment Models	312	<OBJECT Code="*" Type="312"/>	<OBJECT Code="System generated code" Type="312"/>
Prepayment Rules	300	<OBJECT Code="*" Type="300"/>	<OBJECT Code="System generated code" Type="300"/>
Stochastic Rate Index Rules	315	<OBJECT Code="*" Type="315"/>	<OBJECT Code="System generated code" Type="315"/>

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table D-2 (Cont.) Input Values for OBJECTMIGRATION.xml File

Object Name	Object Type Id	Input value in OBJECTMIGRATION.XML for the wild card	Input value in OBJECT_MIGRATION.xml for Object Code
Alternate Rate Output Mapping Rules	402	<OBJECT Code="*" Type="402"/>	<OBJECT Code="System generated code" Type="402"/>
FTP Standard Processes	204	<OBJECT Code="*" Type="204"/>	<OBJECT Code="System generated code" Type="204"/>
FTP Stochastic Processes	203	<OBJECT Code="*" Type="203"/>	<OBJECT Code="System generated code" Type="203"/>
Break Identification Processes	816	<OBJECT Code="*" Type="816"/>	<OBJECT Code="System generated code" Type="816"/>
Cash Flow Edits	700	<OBJECT Code="*" Type="700"/>	<OBJECT Code="System generated code" Type="700"/>
FTP Rate Card Products	605	<OBJECT Code="CONFIG" Type="605"/>	<OBJECT Code="CONFIG" Type="605"/>
Rate Reports	604	<OBJECT Code="*" Type="604"/>	<OBJECT Code="System generated code" Type="604"/>

D.5 Logging

- <War_deployed_location>/logs/offline_migration.log
- <War_deployed_location>/logs/FUSIONSUBSYSTEMSService.log

Copyright © 2011, 2026, Oracle and/or its affiliates.

E

Transfer Pricing Error Messages

The following table lists the standard error messages that are encountered in Funds Transfer Pricing application.

Table E-1 Error Messages in OFS FTP

Error Code	Error Message	Description
105	[Rule_Name]:[Rule_Name] not found in database.	One or more assumption rules chosen in the Process rule are deleted or corrupted in the database.
110	No match for IRC.	There is no IRC selected for the specific product member within the Transfer Pricing rule.
115	No data for IRC.	The IRC selected in the Transfer Pricing rule has no historical rates.
120	Dimension Type mismatch: COL vs. Cnfg ID.	The Product Leaf defined in Application Preferences is different from the Product leaf used in the methodology assignments in the Transfer Pricing rule.
505	Negative [rate type] Rate for Product: [number].	An instrument has been repriced at a negative rate due to the lookup rate + margin is less than 0.
540	Prepayment tier not selected, zero prepayment rate used.	Either Prepayment Model or Prepayment Rate is undefined for a specific Prepayment Rule.
545	The last payment date is before the origination date.	The last payment date is before the origination date for a specific record in the instrument table.
550	No reprice frequency for the adjustable record.	The instrument record is an Adjustable record (adjustable_type_cd <> 0), and has a 0 Reprice Frequency (reprice_freq = 0).
555	No schedule was found for the record.	This error could occur in two different situations: The instrument record has a Schedule Amortization Code (amrt_type_cd = 800, 801, or 802) and has no matching record in the Payment Schedule table (fsi_d_payment_schedule). The Prepayment Method is set to a Prepayment Model, which does not exist or has no dimensions set up.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table E-1 (Cont.) Error Messages in OFS FTP

Error Code	Error Message	Description
560	Pattern data cannot be found for AMRT_TYPE: [number].	An instrument record with Custom Amortization Codes (amrt_type_cd > 999) has no User Defined Payment Pattern defined.
565	No pattern information found for record.	An invalid Adjustable Type Code was found in the instrument record.
570	Maturity date from the record does not match schedule data.	For an instrument record with Schedule Amortization Code (amrt_type_cd = 800, 801, or 802), the remaining number of payments (remain_no_pmt_c) does not match the schedule of payments defined in the Payment Schedule Table (fsi_d_payment_schedule).
575	Conv amrt with interest in advance defaulted to arrears.	An instrument record with one of the following Amortization Codes: 100, 400, 500, 600, 710, must implicitly have an interest in the arrears method. If Interest in advance Interest Type (int_type = 2) is defined for an instrument record with an implicit interest in arrears method, then this error is recorded and the Interest Type is set to arrears (int_type = 1) for cash flow generation purposes.
585	Invalid Amrt.Type detected. Defaulted to Simple Int.	The instrument record has an Amortization Code that is outside the reserved custom range or is nonstandard.
710	Insufficient memory to read[Rule Name]:[Rule Name].	The Historical Rates, Prepayment Rule, Adjustment Rule, and Transfer Pricing Rule are all loaded into memory when processing. If several other applications are active on the machine, there may not be enough available memory to load any or all of these rules.
715	Transfer Pricing Method not implemented.	The transfer pricing methodology selected is listed but not yet available.
725	1st point used.	The origination date and term of the instrument fall before the first date on the IRC.
730	The last point used.	The origination date and term of the instrument are in the future after the last date on the IRC.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table E-1 (Cont.) Error Messages in OFS FTP

Error Code	Error Message	Description
740	TP Rule out of sync on product[number].	The Transfer Pricing Rule still has a methodology defined for a product leaf that is no longer defined in dimension member setup.
745	COL table empty; Dimension Type [member number] used.	Only leaf members with detailed information can be used in the Transfer Pricing Rule. Some key dimensions defined in the Catalog_of_Leaves have associated detailed information tables. This error occurs if a leaf member of a Transfer Pricing Rule does not have associated detailed information, like account type or offset account.
755	Dimension key not found in Migration Table.	The combination of Common_COA_ID and Org_Unit_ID on the instrument data does not exist in the Management Ledger.
760	Negative Transfer Rate.	Negative transfer rates can only occur with a cash flow transfer pricing methodology if the cash flow fields on the instrument have bad data that create negative transfer rates.
775	Invalid TP method used for Management Ledger data: [number].	An invalid transfer pricing method has been assigned to the Transfer Pricing Rule used in the TP Process. There are only six valid methods for use on the Management Ledger table: Moving Averages, Redemption Curve, Spread from Interest Rate Code, Caterpillar, Weighted Average Perpetual, and Unpriced Account.
780	Account type not defined for leaf member [number], Earning asset used.	The account type attribute is not defined for a particular product leaf member.
785	0 Instrument Table rows processed.	There is no instrument data, or a filter is excluding all instrument data defined in the active Transfer Pricing Rule.
790	Invalid cash flows generated for this row or divide by zero.	One or more cash flow fields for detailed instrument data have bad data or missing data.
795	Invalid duration generated for this row.	One or more fields used for duration calculations from the instrument tables have bad or missing data.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table E-1 (Cont.) Error Messages in OFS FTP

Error Code	Error Message	Description
800	Org Unit leaf member not defined, default offset used.	The Organizational Unit number is not defined in Dimension Management.
805	Org Unit offset not defined, default offset used.	The offset unit is not defined in Dimension Management.
810	Divide by zero.	Internal arithmetic calculation error encountered when there is a division by zero.
820	Not an instrument table	The instrument table is missing the required cash flow fields.
840	The calculated average life is 0	Average life for the account is zero, mostly, due to no cash flows.
845	No transfer pricing method defined for PRODUCT: [product], CURRENCY: [currency]	System could not find any transfer pricing method for concerned product.
850	PRODUCT: [product], CURRENCY: [currency] - no Adjustment Method has been defined.	System could not find any Adjustment method for concerned product.
860	Invalid conditional assumption defined for ID number [idnumber] ID_NUMBER, Product [Product id] PRODUCT_ID and currency [currency] ISO_CURRENCY_CD.	Incorrect filter defined for ID_NUMBER, Product [Product id] PRODUCT_ID and currency [currency] ISO_CURRENCY_CD.
865	Adjustment Rule Definition incomplete or missing for Rule Sys ID: [objdefid]OBJ_DEF_ID. Please complete rule definition for all relevant product/currency combinations.	Adjustment rule is not properly defined for OBJ_DEF_ID.
875	No Alternate Rate Output Mapping Rule details found for the selected rule. (Object Definition ID: "[objdefid]OBJ_DEF_ID")	Alternate rule is not defined for concerned rule.
880	Invalid interest rate code [interest rate cd] IRC specified for currency [currency] CURRENCY.	Interest rate code doesn't have proper rates or IRC format is incorrect.
900	None of the account tables selected in the Transfer Pricing Process Rule have the Break Funding classification, so Break Funding calculations were not run.	No break funding table was selected, so break calculations didn't run.
905	Transfer Pricing method [calc meth cd] METHOD used with product[product] PRODUCT is not appropriate for instruments in table [table name] TABLE_NAME. Cause:[table name] TABLE_NAME is not registered with the appropriate columns.	Incorrect transfer pricing method found for instruments table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table E-1 (Cont.) Error Messages in OFS FTP

Error Code	Error Message	Description
915	Negative Strips are adjusted.	Reduced portfolio balance is adjusted via negative strips.
920	The selected TP Charge/Credit Balance option was not available in the Ledger Table. Defaulting to Ending Balance.	selected TP Charge/Credit Balance option was not available in the Ledger Table. Defaulting to Ending Balance.
925	No All-In TP Rule has been defined.	All in TP formula is not defined for selected product.
930	Leaf Member accrual basis attribute not defined for Product ID:[product id]. Defaulted to 30/360.	Accrual Basis is not defined for selected product.
935	LRD_BALANCE = 0 and Adjustable Type Code > 0. Defaulting LRD_BALANCE = CUR_PAR_BAL	For Floating rate instrument LRD Balance = Cur Par Bal.
940	LRD_BALANCE and CUR_PAR_BALANCE = 0, Adjustable Type Code > 0. Unable to Process this instrument.	As LRD and Current Balance both are zero for floating rate instrument, System can't process the record.
945	Dimension members not found in required format. Reverse Population has not been run for the selected FTP Product Hierarchy: [prod hier id]	Dimension members not found in required format.
955	Product :[product] is not a commitment record and the instrument table does not contain the required Commitment Date columns. The Adjustment Rule - Assignment date has been defaulted to As of Date.	Assignment date is defaulted to as of date, as didn't find any commitment columns for selected record.
960	Transfer Rate is blank for the ID_NUMBER :[id_number]. Unable to Process Rate Lock Option Cost.	Didn't find any transfer rate, record skipped from rate lock option cost calc.
1010	Dimension members not found. Please confirm that Reverse Population has been run for the FTP Product Hierarchy used by the FTP Adjustment Calculation process.	Dimension members not found. Please confirm that Reverse Population has been run for the FTP Product Hierarchy used by the FTP
1020	Invalid Amortization Type Code encountered. Defaulting AMRT_TYPE_CD = 700, non-amortizing instrument.	Invalid Amortization Type Code encountered.
1030	No holiday code found while calculating accrual factor, holiday code = 0. please check.	No holiday code found while calculating accrual factor, holiday code = 0.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table E-1 (Cont.) Error Messages in OFS FTP

Error Code	Error Message	Description
1040	During migrate/direct tp of ML, error occured while inserting FSI_D_MANAGEMENT_LEDGER, please check.	During migrate/direct tp of ML, error occured while inserting FSI_D_MANAGEMENT_LEDGER,
1050	No Direct TP method defined on leaf and No migration option for Table :FSI_D_MANAGEMENT_LEDGER to use.	No Direct TP method defined on leaf and No migration option for Table :FSI_D_MANAGEMENT_LEDGER to use.

Copyright © 2011, 2026, Oracle and/or its affiliates.

F

Non-Maturity Products Model Analysis

Exploratory Data Analysis – At this stage, User can pick the historical data which will be used for model analysis in subsequent steps.

This UI lets you define as of date range, for which historical data needs to be picked. As Of Date (Frequency) – This helps you define data points frequency, which could be daily, monthly, bi-weekly etc. More frequent/granular data, helps you enhance your model quality. Here, you need to make sure not to select frequency which is less than frequency at which historical data is available. So, if you have loaded monthly as of dates as per every month end run, then selected frequency for this field must be at least monthly or more than this, could be quarterly, Semi-Annually or Annually. This is required as there are models in subsequent steps like Drawdown Analysis, GBM or SARIMAX which needs series of data points for given as of date range.

Moving Average Period – This defines the period over which moving average gets calculated to create Bollinger band. Sigma Co-efficient helps to define the lower and upper bands for Bollinger band graphs.

Pass-Through rate Calc

Pass Through Rate is based on linear regression between market and offered rate. For current accounts, offered rate can be zero, in those scenarios, pass through rate will be zero, as any change in market rate doesn't have any impact on offered rate.

Stable Balance Calculations

Drawdown Analysis: This model does calculations based on maximum drawdown between two data points for given period. Say analysis is done over past 5 years and monthly data points are given. On each data point, balance is available, system calculates balance reduction for each data point from previous data point. Finally, whatever will be the maximum balance reduction in two adjacent data points that is subtracted from point in time balance (Balance on latest as of Date) and remaining balance is the stable balance.

Geometric Brownian Motion – This model works based on historical volatility and confidence interval selected in the UI. It assumes underlying data follows normal distribution, and as per the confidence interval, X% of the balance falls between below ranges; which helps the system derive stable balance.

95% Confidence Interval – (-1.96 to +1.96)

90% Confidence Interval – (-1.645 to +1.645)

80% Confidence Interval – (-1.282 to +1.282)

Here User has the option to choose outlook scenario factor, which will be multiplied with historical volatility and subsequently used for GBM based stable balance calculations. GBM outcome is based on both portfolio level balance volatility and account level balance volatility. User can select any of the stable balances, which are more suitable as per the given context.

Decay Rate Calculations

Vintage Analysis: The vintage run-off model seeks to segregate deposit balances based on historical tranches called vintages. A vintage consists of all of the individual accounts for a non-

Copyright © 2011, 2026, Oracle and/or its affiliates.

maturing deposit account type that are opened within a particular time bucket; it could be month, week. Behavioral characteristics are applied to each vintage by calculating the monthly decay within each vintage. Run off calculations are done across data points and across account origination buckets and finally an average is taken across both dimensions to arrive at final decay rate for the selected portfolio.

Vintage analysis mandatorily needs account level data to segregate accounts by their origination dates, in case account level data is not available, there is a checkbox given in UI, which can skip this model.

SARIMAX: Among the various approaches available, the SARIMAX (Seasonal Autoregressive Integrated Moving Average + exogenous variables) model stands out as a powerful tool for modeling and forecasting both trends and seasonal variations in temporal data, while incorporating exogenous variables into the analysis to improve prediction accuracy. This model belongs to ARIMA models.

It is based on three key components: autoregression (AR), moving average (MA) and integration (I).

Autoregression (AR) considers past values of the time series to predict current values. The moving average (MA), on the other hand, tackles past errors in predictions. The moving average consists of performing a linear regression on the last q error values to predict the current value.

When temporal data show seasonal variations, the SARIMA model takes over the scene. The term “Seasonal” is added to ARIMA to indicate that this model can capture patterns that repeat at regular intervals. Based on the decay rate in each of the periods, weighted average life is calculated by multiplying run-offs matrix by time series.

GBM (Geometric Brownian Motion): Conceptually, this is same as mentioned above under stable balance, it is extended here to capture forecasted balance and how those deteriorate over forecasted period. Using regular run-offs in each period and multiplying those with period itself, weighted average life is calculated.

If there is any regulatory cap on WAL of non-maturity accounts like CASA, then system automatically perform iteration by changing confidence interval to keep WAL under the given regulatory cap, and accordingly produces decay rate profile as well.

Copyright © 2011, 2026, Oracle and/or its affiliates.

G

Process Tuning

This appendix provides information on configuring certain Oracle Financial Services applications for application multiprocessing through the Process Tuning User Interface.

> **Note**
>
> The Process Tuning User Interface is available for definition ONLY for a User who has a role mapping of 'Administrator' or 'Auditor'. The 'Auditor' role mapped user has 'READ-ONLY' access to the Process Tuning UI.

Both the Asset Liability Management (ALM) and Funds Transfer Pricing (FTP) applications involve processing large quantities of data and the goal of all customers is to reduce processing time so they have more time to focus on the analysis of the results. The process execution window is closely related to the volume of data it operates on, underlying metadata definitions, in-memory calculations, processing individual calculation elements or selections (for example, transfer pricing method, propagation, account-level cash flows, financial elements), and so on. To achieve optimal performance, application Process Tuning allows application administrators to specify the number of concurrent sub-processes to launch for each application process that is initiated and also allows users to define how to slice the data for each sub-process. This process is described in detail below.

The following are basic guidelines, which must be considered to achieve optimal performance:

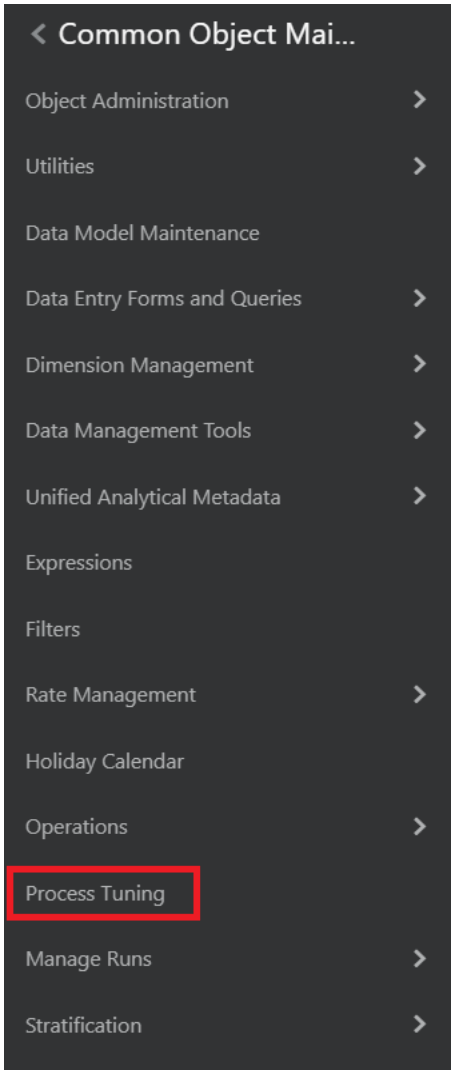
- Application Process Tuning
- Operating System-Level Tuning
- Database-Level Tuning
- Defining Multiprocessing Parameters
- Application Multiprocessing Model
- Multiprocessing Options
- Tuning Application Multiprocessing
- Performance Tuning Applicable to FTP
  For the OFS FTP application, the following types of Performance Tuning are applicable:
- Operating System-Level Tuning
  The operating system level performance tuning is related to Semaphores and shared memory settings.

G.1 Defining Multiprocessing Parameters

Access the multi-processing/tuning UI under the **Common Object Maintenance** on the **LHS** menu.

Copyright © 2011, 2026, Oracle and/or its affiliates.

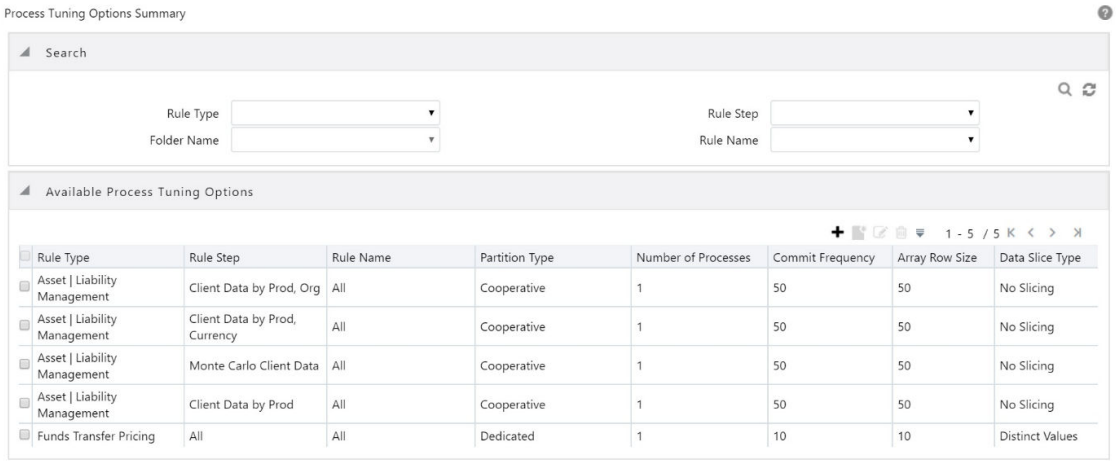
Figure G-1 Navigation to Process Tuning UI



This opens the Process Tuning Options Summary window which is the gateway to perform all the required Process Tuning Options-related functionality.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure G-2 Process Tuning Options Summary page



The application installation process seeds default multiprocessing parameters. By default, multiprocessing is disabled for all applications. It can be enabled by inputting specific settings for the following applications:

- Asset Liability Management – Deterministic and StochasticProcessing
- Funds Transfer Pricing – Standard and StochasticProcessing

Follow these steps to define multiprocessing parameters :

1. From **Process Tuning Options Summary** page click **Add** to create a new Process Tuning assumption. You can create only one process tuning assumption per combination of Rule Type and Rule Name.
2. Select the Assignment LevelDetails:
   - Folder Name
   - Rule Type
   - Rule Step
   - Rule Name
3. Define multi processing parameters:
   - PartitionType
   - Array Row Size
   - Number of Processes
   - Commit Frequency
4. Select the Data slice type
5. Select Data slicing columns

Details for each step are explained in the following sections:

- Assignment Levels Details
- Parameter Specification
- Multiprocessing Options
- Engine Overrides

Copyright © 2011, 2026, Oracle and/or its affiliates.

For each step, the relevant multiprocessing parameters are described. Some applications override the configuration to handle special processing conditions. The Engine Overrides section explains these special processing conditions.

- Assignment Levels Details
- Parameter Specification
- Multiprocessing Options

G.1.1 Assignment Levels Details

Multiprocessing parameters can be specified at different levels. A Multiprocessing Assignment Level is the category of ALM or FTP execution that is processed with a designated set of multiprocessing parameters.

OFSAA supports multiprocessing assignments at multiple levels. Because an application allows configurations that mix these levels, it is important to understand the order in which the application resolves the multiprocessing parameters when the configuration mixes assignment levels. The order is as follows:

- Specific step of a specific process for anengine
- All steps of a specific process for anengine
- Specific step of all processes for anengine

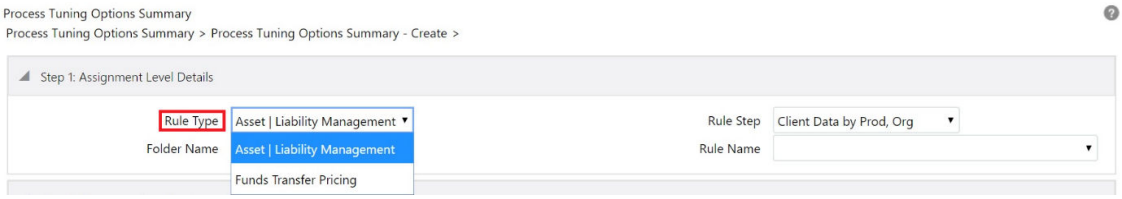
Multiprocessing assignments can be done at the following levels:

- RuleType
- RuleStep
- RuleName

Rule Type

When specifying multiprocessing parameters at the Rule Type level, all Rule Names for that Rule Type are processed with the designated parameters.

Figure G-3 Rule Type drop-down on Process Tuning Option Summary page



The following table describes the valid Rule Type values present in the **Rule Type** drop-down list

Table G-1 Rule Types

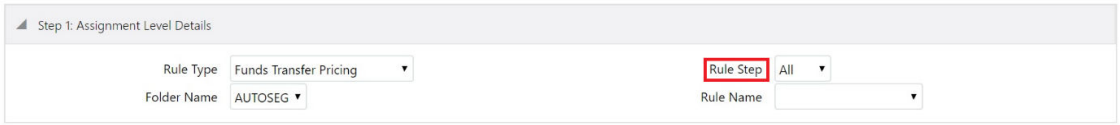
Process Engine Code	Rule Type	Description
2	Asset | Liability Management	Asset | Liability Management Process
3	Funds Transfer Pricing	Funds Transfer Pricing Process

Copyright © 2011, 2026, Oracle and/or its affiliates.

Rule Step

The Rule Step identifies a particular phase of an OFSAA process. They are reserved names specific to each Rule Type.

Figure G-4 Rule Step drop-down on Process Tuning Option Summary page



Rule Step is a drop-down list that is filtered based on its mapping to the selected Rule Type. The following table provides the valid combinations of Rule Type and Rule Step:

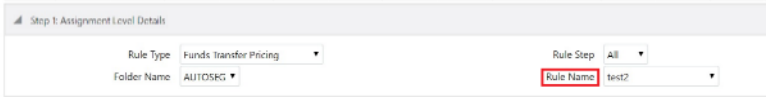
Table G-2 Rule Types

Process Engine Code	Rule Type	Rule Step
2	Asset | Liability Management	Client Data by Prod
2	Asset | Liability Management	Client Data by Prod, Org
2	Asset | Liability Management	Client Data by Prod, Currency
2	Asset | Liability Management	Monte Carlo client data
3	Funds Transfer Pricing	All

Rule Name

Users can optionally specify multiprocessing parameters at the Rule Name level to override any parameters assigned at the Rule Type or Step level. This enables you to individualize your multiprocessing options to handle situations unique to specific processes. In most cases, defining multi-process assumptions at the Rule Type or Step level is sufficient.

Figure G-5 Rule Name drop-down on Process Tuning Option Summary page



The**RuleName** drop-down list displays the list of processes defined in Asset Liability Management or Funds Transfer Pricing applications within the specified folder. This is controlled by the value selected in **RuleType**.

G.1.2 Parameter Specification

In this step, the following parameters are associated with Rule Types and/or Rule Names:

- PartitionType
- Number ofProcesses
- Array Row Size and CommitFrequency

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure G-6 Parameter Specifications on Process Tuning Option Summary page



Partition Type

Partition Type selection is required when table partitioning is used within your instrument tables. This drop-down list displays the following values:

Table G-3 Partition Codes

Partition Type Code	Description
0	Single
1	Cooperative
2	Dedicated

If no partitioning is required, then the default selection of “Single” should be selected. If partitioning is required, then users should select either Dedicated or Cooperative. For more information on these selections, refer to the Multiprocessing Options > Unit of Work servicing section

Number of Processes

The number of process selection indicates the number of concurrent subprocesses to be launched by the engine. Typically, the number of subprocesses should be less than or equal to the number of CPUs available on the server. Experiment with the number of processes setting to find the optimal value.

Array Row Size and Commit Frequency

The number in the Array Row Size column sets the number of updates sent to the database. Commit Frequency is the number of rows after which the process commits changes.

The array update enables users to specify the number of rows updated using a single cursor. Array updating accumulates updated rows during processing until the defined update frequency is reached and then sends the updates to the database for processing.

You must do some trials to set these at the optimal level. Commit frequency can be set as 999 and arraysize rows as 500. Setting the array update size to be greater than the commit frequency does not affect because the array update size is limited by commit frequency.

Oracle recommends the following to set commit frequency forALM:

- If only **Process Cash Flow** writing is applicable as part of process definition then use Commit Frequency optimally greater than 0.
- If **Process Cash Flow** writing is not applicable but output to `RES_DTL, CONS_DTL`,and so on as part of process definition then the Commit Frequency should be set to 0 if the commit is required at end of unit-of-work.
- If both **Process Cash Flow** and output to `RES_DTL, CONS_DTL`are applicable then set Commit Frequency greater than 0 to see the benefits of Process Cash Flow Bulk Processing. If the commit is required at end of unit-of-work then set it to 0.

Copyright © 2011, 2026, Oracle and/or its affiliates.

G.1.3 Multiprocessing Options

Multiprocessing options include the following selections:

- Data slice type
- Data slice columns

Data Slice Type

Figure G-7 Data Slice Type Selection on Process Tuning Option Summary page



Data Slice Type is a static drop-down list with two values. You should choose distinct values to enable multiprocessing. After choosing distinct values, you must further select their Data Slicing columns.

Table G-4 Data Slice types

Data Slice Type Code	Description
1	Distinct Values
2	No Slicing

Data Slicing Columns

Choose the edit option to open the Data Slicing Columns window to select the slicing columns.

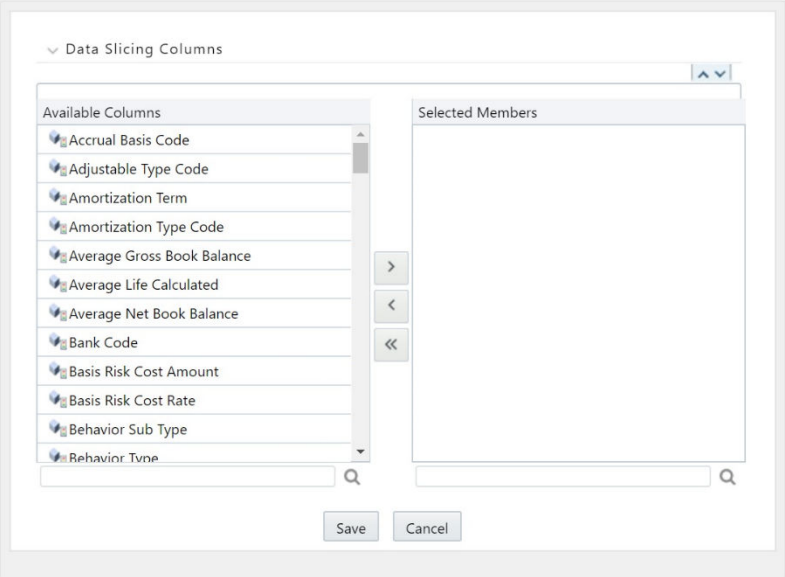
Figure G-8 Data Slicing Columns on Process Tuning Option Summary page



This window contains all numeric columns of portfolio table class. The columns with null values cannot be used for data slicing

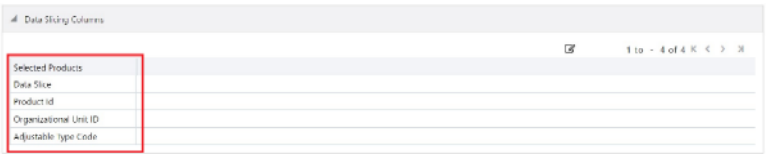
Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure G-9 Data Slicing Column selection



Columns selected in the pop-up window are displayed on the main page

Figure G-10 Selected Data Slicing Columns



Columns should be selected so that the instrument data can be broken down into equal segments to efficiently utilize the maximum amount of processing power. It is noted that using dimensions like the product, organization unit, and so on do not always lead to creating equal data segments. To address the requirement for uniformity in creating optimal data segments, a seeded dimension ‘Data Slice’ is available in both ALM and FTP that can be used to distribute instrument data evenly and balance the load across all active workers for certain types of processes.

For more information on using the Data Slice dimension, see the Data Slicing section of OFSAA Data Model Utilities Guide.Also, see the Engine Overrides section.

Engine Overrides

For certain conditions, the OFSAA processing engines override the multiprocessing definition for an assignment level. The overrides are as follows:

Funds Transfer Pricing

Funds Transfer Pricing configures the data slicing columns automatically using the Product Leaf Column defined in the active Application Preferences as the default slicing column for all runs. Different steps in the same processing run can use additional slicing columns. Bulk and propagation calculation steps, as well as Non-Cash Flow and Management Ledger pricing/ migration runs, use the `ORG_UNIT_ID` column as an additional slicing column. The Cash Flow Transfer Pricing step also uses the `ORG_UNIT_ID` column if not combined with Option Cost

Copyright © 2011, 2026, Oracle and/or its affiliates.

Calculations. For Option Cost Calculation, the engine employs ORIGINATION_DATE as the primary slicing column with the Product Leaf Column as secondary.

Asset Liability Management

Asset Liability Management configures the data slicing columns automatically using the Product Leaf Column defined in the active Application Preferences as the default slicing column for all runs. The Asset Liability Management engine adds more slicing columns based upon the parameters specified in the Asset Liability Management Process, as follows:

- If the **Product/Organizational Unit functional dimensions** are selected, the engine adds ORG_UNIT_ID as an additional slicing column.
- If the **Product/Currency functional dimensions** are selected, the engine adds ISO_CURREN- CY_CD as an additional slicing column
- If the **Product/Organizational/Currency functional dimensions** are selected, the engine adds ORG_UNIT_ID and ISO_CURRENCY_CD as additional slicing columns

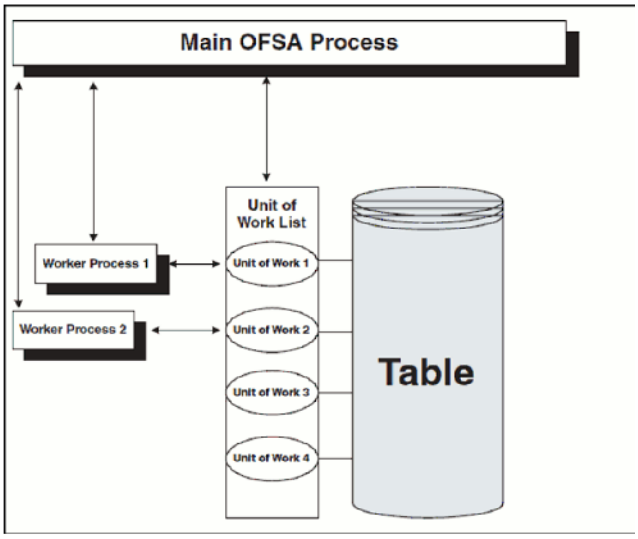
G.2 Application Multiprocessing Model

OFSAA application multiprocessing is based on the concept of a unit-of-work. A unit-of-work is a set of rows from the database. A single OFSAA process becomes multiple processes by dividing the single process according to distinct sets of rows. Units of work are distributed to worker processes until all processes are completed. To achieve multiple parallel processes, the following options must be configured:

- Creating a list or lists of units ofwork
- Defining the number of worker processes to service the units-of-worklists
- Defining how the worker processes service the unit-of-worklists

The specifics of each option are discussed. The following diagram illustrates the basic multiprocessing principles:

Figure G-11 Basic Multiprocessing principles



1. The main process makes a list of all units of work that must beprocessed.

Copyright © 2011, 2026, Oracle and/or its affiliates.

2. The main process spawns worker subprocesses. Each worker subprocess is assigned a unitof work by the main process.
3. When all units of work have been completed, the worker subprocess exits and the mainprocess finishes any clean-up aspects of processing.
4. During processing the following istrue:
   - Each worker subprocess must form its databaseconnection.
   - A unit of work is processed only by a single workersubprocess.
   - Different units of work are processed at the same time by different workersubprocesses.

If data is not distributed well across physical devices, I/O contention may offset the advantage of parallelism within OFSAA for I/O bound processing. Oracle recommends that users choose dimensions or combinations of dimensions that produce an equal distribution of data records.

G.3 Multiprocessing Options

The Multiprocessing Options are the settings and parameters that control how individual ALM or FTP Processes are processed by the Cash Flow engine. The seeded data model includes default settings for all of the multiprocessing options, but you can also customize these settings for your use through the process tuning user interface as described above. This section provides more details for the different Multiprocessing options as further background. These options are:

- Units ofWork
- Unit-of-WorkServicing
- WorkerProcesses

Units of Work

The OFSAA processing engines determine units of work for any job based upon the Process Data Slicing Code (`PROCESS_DATA_SLICES_CD`)assignment. The Data Slicing Code includes one or more columns by which data in the (processing) table is segmented. The individual segments are the defined Units of Work.

The Process Tuning user interface enables you to specify different unit-of-work definitions for your processes. You can specify one unit-of-work definition for one set of processes and then specify a different unit-of-work definition for another set of processes.

The OFSAA Processing Engines determine the units of work for a job by executing the following statement (with filtering criteria applied) on every table the process is run against:

```
select distinct<data slice columns> from <table> where <filter condition>;
```

where <data slice columns>is the comma-separated list of columns used for data slicing, <table>is the name of the instrument table being processed, and <filter condition>is the additional filter (if any) for the process. Any numeric column or columns in a portfolio table can be used for data slicing.

Unit-of-Work Servicing

Unit-of-work servicing identifies how the OFSAA processing engines interact with Oracle Database Table Partitioning.

What is Partitioning?

Copyright © 2011, 2026, Oracle and/or its affiliates.

Partitioning addresses the key problem of supporting large tables and indexes by enabling you to decompose them into smaller and more manageable pieces called partitions. After partitions are defined, SQL statements can access and manipulate the partitions rather than entire tables or indexes. Partitions are especially useful in data warehouse applications, which store and analyze large amounts of historical data.

What Is Unit-of-Work Servicing?

Unit-of-work servicing specifies how individual units of work are processed for a table thatis partitioned.

For a partitioned table, an application Rule Type or Rule Step can create multiple unit-of-work lists by executing the following statement (with filtering criteria applied) on every table partition the processis run against:

```
select distinct < data slice columns > from < table_partition_n >;
```

For example,

```
SELECT DISTINCT REV.COLUMN_NAME, MLS.DISPLAY_NAME from REV_TAB_COLUMNS REV,
REV_TAB_COLUMNS_MLS MLS
WHERE REV.TABLE_NAME = MLS.TABLE_NAME AND REV.COLUMN_NAME = MLS.COLUMN_NAME
AND REV.TABLE_NAME = 'PORTFOLIO' AND REV.DATA_TYPE = 'NUMBER'
AND MLS.MLS_CD = 'US' UNION
SELECT 'DATA_SLICE_ID' COLUMN_NAME, 'Data Slice' DISPLAY_NAME
from DUAL UNION
SELECT 'FINANCIAL_ELEM_ID' COLUMN_NAME, 'Financial Element' DISPLAY_NAME
from DUAL
ORDER BY DISPLAY_NAME;
```

where <data slice columns>is the comma-separated list of columns used for data slicing. Any column or columns in a table can be used for data slicing. <table_partition_n> are the unique table partitions of a table where n is assumed to be greater than 1.

The different Servicing methodologies provided in OFSAA are listed in the following table. You cannot add any customized Servicing methodologies.

Table G-5 Partition Type Codes

Partition Type Code	Description
0	Do not use partitions (Single Servicing)
1	Use shared partitions (Cooperative Servicing)
2	Use non-shared partitions (Dedicated Servicing)

Single Servicing

Single Servicing indicates that the processing engine fulfills unit-of-work requests regardless of any table partitioning. As each process completes, it requests the next unit-of-work segment, whether that segment belongs in the same Table partition.

Use Single Servicing when you do not have Oracle Table Partitioning in your database.

Cooperative Servicing

Cooperative Servicing indicates that the processing engine fulfills unit-of-work requests so that each process works against a specific partition unless it is idle. Idle processes then work against the next available unit-of-work segment, whether that segment belongs in the same Table partition.

Copyright © 2011, 2026, Oracle and/or its affiliates.

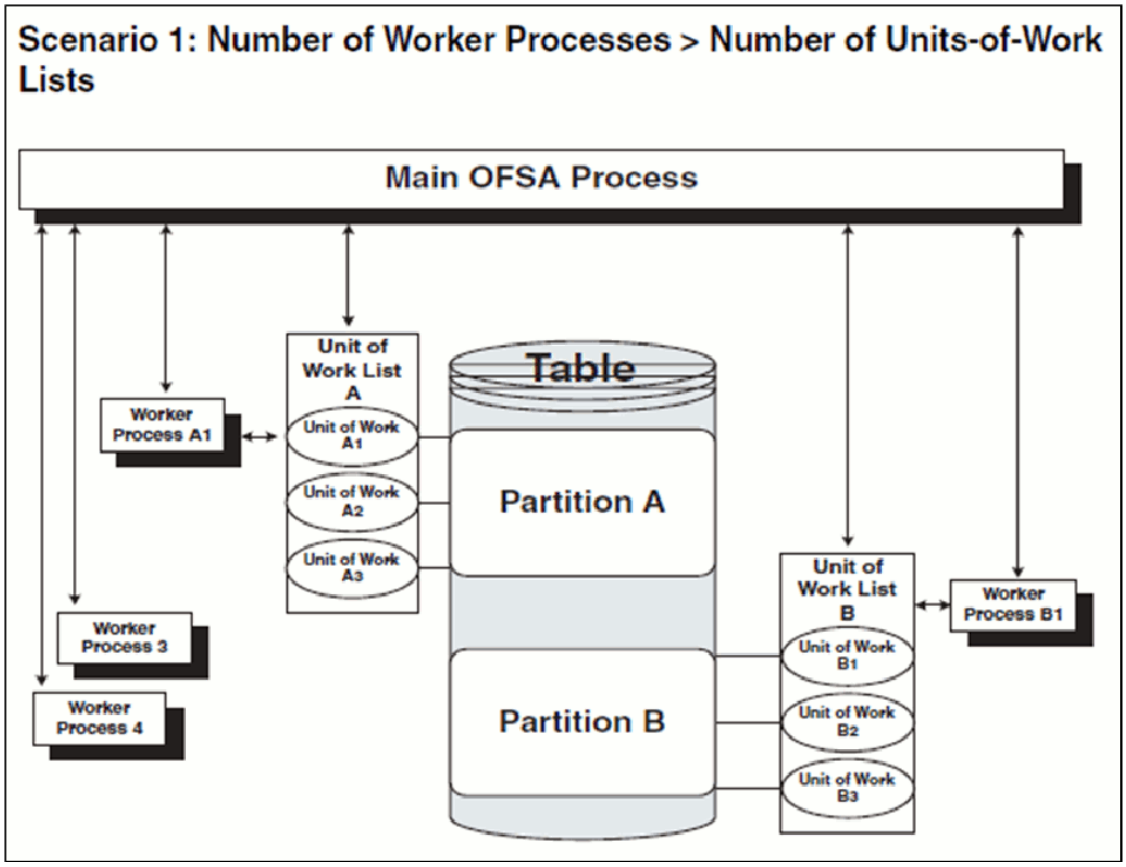
Dedicated Servicing

Dedicated Servicing indicates that the processing engine fulfills unit-of-work requests so that each process works against a specific partition.

Examples of How Worker Processes Service Units of Work

Partition Type code defines how worker processes service the unit-of-work list or lists. Partition Type ‘Single’ results in a single unit-of-work list. All available worker processes service the list until all units of work are complete. When the Partition Type is Cooperative or Dedicated and the table to be processed is partitioned, multiple units of work lists are created. The following scenarios explain how the worker processes service multiple units-of-work lists

Figure G-12 How the worker processes service multiple unit-of-work lists

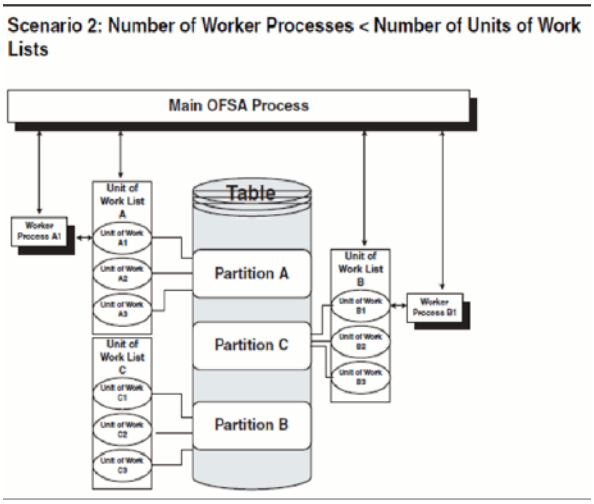


- The main process makes two lists of all units of work that need to be processed, unit-of-work list A and unit-of-work list B, respectively. The assumption setup is that the Table has two partitions.
- The main process spawns four worker processes. A dedicated worker process is assigned to ser- vice each units-of-work list, Worker Process A1, and Worker Process B1 respectively. (The setup is that the Number of Processes is4.)
  - If Partition Type is Cooperative, Worker Process 3 and Worker Process 4 assist Worker Process A1. When a unit-of-work list is complete, the available worker processes assistthe dedicated worker process on their unit-of-work list.
  - If Partition Type is Dedicated, Worker Process 3 and Worker Process 4 do not assistthe dedicated worker processes.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- When all units of work are completed, the worker process exits and the main processfinishes any clean-up aspects of processing
- During processing the following istrue:
  - Each worker process must form its database connection.
  - A unit of work is processed only by a single worker process.
  - Different units of work are processed at the same time by different worker processes

Figure G-13 How different units of work are processed at the same time by different worker processes



- The main process takes three lists of all units of work that need to be processed: unit-of-work list A, unit-of-work list B, and unit-of-work list C, respectively. (The setup is that the Table has threepartitions.)
- The main process spawns two worker processes. A dedicated worker process is assigned to ser- vice a units-of-work list, Worker Process A1, and Worker Process B1 respectively. (The setup is that the Number of Processes is 2.)
  - If Partition Type is Cooperative, Worker Process A1, and Worker Process B1 work untilall units of work are complete from all three unit-of-work lists.
  - If Partition Type is Dedicated, the first worker process to complete their unit-of-work list services unit-of-work list C. When the other worker process completes their list, theworker process exits.
- When all units of work are completed, the worker process exits and the main processfinishes any clean-up aspects of processing.
- During processing the following istrue:
  - Each worker process must form its databaseconnection.
  - A unit of work is processed only by a single workerprocess.
  - Different units of work are processed at the same time by different workerprocesses.

WorkerProcesses

Worker Processes refer to the number of individual processes working simultaneously to complete the job. The Main OFSAA Process launches individual worker processes. OFSAA enables you to specifythe number of worker processes for your jobs.

Copyright © 2011, 2026, Oracle and/or its affiliates.

G.4 Tuning Application Multiprocessing

Tuning for optimal application multiprocessing settings is an exercise similar to tuning a database. It involves experimenting with different settings under different load conditions.

Database Bound versus Engine Bound Jobs

OFSAA jobs fall into the following two categories:

- Database bound: The jobs that spend more time within databasemanipulations.
- Engine bound: The jobs whose calculations are complex, with the time spent withdatabase operations being small compared to the amount of time doing calculations.

The following table lists jobs by Application and identifies whether the job is usually database bound or Engine bound.

Table G-6 Lists jobs by Application and identifies

Application	Job Type	Generic Job Type	Engine / DB Bound	Multiprocessing Enabled	Comments
Funds Transfer Pricing	Ledger Migration	Bulk	DB	Yes	
Funds Transfer Pricing	Bulk Transfer Pricing	Bulk	DB	Yes	Bulk methods defined via conditional assumptions will be handled row by row
Funds Transfer Pricing	Non-Cash Flow Transfer Pricing	Row by Row	DB	Yes	
Funds Transfer Pricing	Cash Flow Transfer Pricing	Row by Row	Engine	Yes	
Funds Transfer Pricing	LEDGER Migration	Row by Row	DB	Yes	
Asset | Liability Management	Detail Processing (Current position, Gap, Market Value)	Row by Row	Engine	Yes	All processing except Formula Results and Auto Balancing
Asset | Liability Management	Formula Results	Row by Row	Engine	No	
Asset | Liability Management	Auto Balancing	Row by Row	Engine	No	

The scalability of database-bound jobs is largely determined by the size of the database server. The scalability of Engine bound jobs is determined by the size of the application server.

Tuning OFSAA processes from the Application Layer

Despite the many multiprocessing options, tuning the OFSAA database from the application layer is achieved by the following process:

- Identify the OFSAA job types that are used by yourorganization.
- For each job type, time the runs for a series of Number of Processessettings
- Based on the results, determine the optimal setting per application.

Copyright © 2011, 2026, Oracle and/or its affiliates.

G.5 Performance Tuning Applicable to FTP

For the OFS FTP application, the following types of Performance Tuning are applicable:

- Management Ledger Updating
- Cursor Sharing Session Parameters

Management Ledger Updating

Within Funds Transfer Pricing, both Ledger migration and direct Transfer Pricing against the Management Ledger table update the Management Ledger table using inserts. This prevents OFSAA from performing Management Ledger updates in parallel. When the Management Ledger is updated (either because the Management Ledger buffer has filled or the process has ended), updating is done by only one process. All other processes must wait for updating to be completed. As the ratio between rows written to the Management Ledger and rows read from Management Ledger increases, the time spent writing to the Management Ledger dominates the time spent reading, resulting in drastically reduced scalability.

Cursor Sharing Session Parameters

You can set the Cursor Sharing database parameter at a session (FTP Application) level. This parameter, when set to **FORCE,** is observed to significantly improve FTP Processing times and Oracle recommends enabling this option.

These are not hardcoded in the engine. You must provided the right values. The Engine does not validate these values. By default, cursor sharing mode is disabled in the ofstp.ini file which is present in `$FIC_DB_HOME/conf` folder:

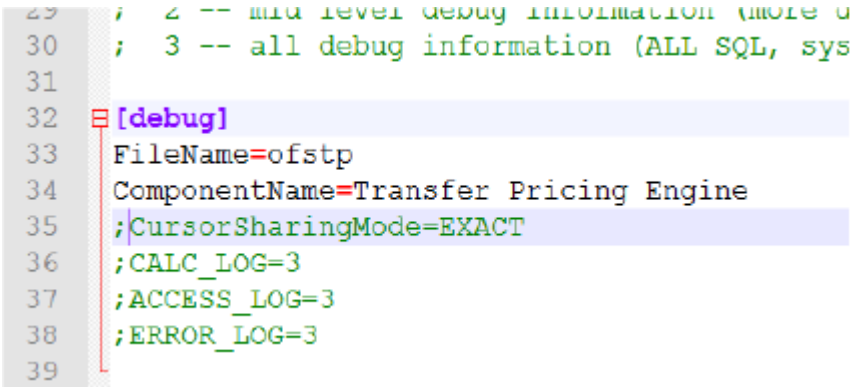
When enabled, the Funds Transfer Pricing engine will read the CursorSharingMode parameter andset `CURSOR_SHARE` at the database session-level accordingly. The FTP Engine will execute the following statement if a value is given for **CursorSharingMode**.

```
ALTER SESSION SET CURSOR_SHARING = <Input Value in upper case>

<Input Values> can be: EXACT, FORCE, SIMILAR
```

To enable this, remove the semicolon in the `ofstp.inifile` as shown :

Figure G-14 Remove ofsfp.in



Copyright © 2011, 2026, Oracle and/or its affiliates.

G.6 Operating System-Level Tuning

The operating system level performance tuning is related to Semaphores and shared memory settings.

- **Semaphores**: Linux Semaphore settings must be set to optimal under the SEMMSL configuration value.
- **Shared Memory**: ALM Engine consumes shared memory during any ALM process execution hence it is crucial to check the availability of current shared memory and tune it accordingly. Shared memory kernel parameters must be set to optimal. The configuration file (ofsrm.ini).

Sometimes users may observe an out of shared memory error reported in ALM Engine log files during process execution. In such cases, one needs to verify and update shared memory size by updating the SharedMemory parameter that can be found in the ofsrm.ini file which is located in $FIC_DB_HOME/conf. The default value of this parameter is 16500 (KB).

- Database-Level Tuning
  Database Level performance tuning describes a group of activities used to optimize and standardize the performance of a database. This can be improved using the following options:
- Recommendations for Performance Improvement
  While configuring an ALM process users can choose to output detailed cash flows (either all or partial). This data is stored in the FSI_O_PROCESS_CASH_FLOWS table. When a process is re-run either for the same as of the date or for a different date, existing data for that process is first deleted before new calculations begin. When the volume of data is high this can take time thereby impacting overall performance. The following recommendations will help improve this.

G.6.1 Database-Level Tuning

Database Level performance tuning describes a group of activities used to optimize and standardize the performance of a database. This can be improved using the following options:

- Simple View versus Materialized View
- Cursor Sharing
- Indexes Optimization

Simple View versus Materialized View

In case ALM processes a huge volume of source records; one may observe a significant number of hits to the views created on REV registration tables (for example, REV_TAB_COLUMNS_V) which result in generating a lot of I/O and cluster waits, in turn spending most of the process execution time spent querying these views only. In such a case, users can change the view to a materialized view. However, for any future upgrade/install customers need to take care of maintaining the changes post-installation.

Cursor Sharing

Cursor handling within the ALM Engine can be implemented using the CursorSharingMode entry in the ofsrm.ini file as mentioned below. It is observed that setting this value to "FORCE" results in the best performance.

Copyright © 2011, 2026, Oracle and/or its affiliates.

1. Update the CursorSharingMode entry in the ofsrm.ini file (File Location: `$FIC_DB_HOME/conf`) with cursor sharing parameter (EXACT, FORCE, SIMILAR).
2. Engine reads the value against CursorSharingMode and sets CURSOR_SHARE at DB Session level accordingly.
3. The cursor level setting is applicable only for the process or session hence the Engine executes the below statement if a value is set against the CursorSharingMode. ALTER SESSION SET CURSOR_SHARING = <Input Value in upper case>

   ```
   <Input Values> can be: EXACT, FORCE, SIMILAR
   ```

4. If no value is supplied, then the Engine will not alter cursor sharing for that session or process.

Process Cash flows Indexes Optimization

While persisting cash flows, the existing unique index on FSI_O_PROCESS_CASH_FLOWS can be dropped and an index on RESULT_SYS_ID must be created before running an ALM process. Dropping the index on FSI_O_PROCESS_CASH_FLOWS is recommended when outputting cash flows for a large number of instrument records. After the process completes, the index must then be added back.

G.6.2 Recommendations for Performance Improvement

While configuring an ALM process users can choose to output detailed cash flows (either all or partial). This data is stored in the FSI_O_PROCESS_CASH_FLOWS table. When a process is re-run either for the same as of the date or for a different date, existing data for that process is first deleted before new calculations begin. When the volume of data is high this can take time thereby impacting overall performance. The following recommendations will help improve this.

1. Create a non-unique index on column RESULT_SYS_ID of the table FSI_O_PROCESS_CASH_FLOWS. The index is expected to make DELETE of existing data faster when a process is re-run.
2. FSI_O_PROCESS_CASH_FLOWS can be partitioned by RESULT_SYS_ID. Automatic List Partitioning can be used so that partitions get added automatically when a new process is defined and executed.
3. Before executing an ALM process, existing data for that process can be removed from FSI_O_PROCESS_CASH_FLOWS by using the DELETE of TRUNCATE PARTITION function. This will ensure the engine does not spend time doing that activity.

Some other server and database level settings that can improve performance are:

1. REDO Log size of the database can be increased. This will reduce log switch waits and improve overall time for analytical operations.
2. CURSOR_SHARING parameter of the database can be set to FORCE. By doing this the database will try to convert SQL queries generated by the cash flow engine to use bind variables which will reduce hard parses and can improve performance.

The recommended settings are just for guidance and you must consult with your database administrators and other system personnel to implement the above, taking your specific infrastructure and other requirements into consideration.

Copyright © 2011, 2026, Oracle and/or its affiliates.

H

Batch to Load the INSTRUMENT_SUMMARY_TABLE

<INFODOM>_LOAD_INSTRMNT_SUMMARY: This seeded batch contains the following DT to load the FSI_D_INST_SUMMARY from individual INSTRUMENT tables:

- FSI_D_MUTUAL_FUNDS
- FSI_D_OTHER_SERVICES
- FSI_D_RETIREMENT_ACCOUNTS
- FSI_D_TERM_DEPOSITS
- FSI_D_TRUSTS
- FSI_D_ANNUITY_CONTRACTS
- FSI_D_BORROWINGS
- FSI_D_CASA
- FSI_D_CREDIT_CARDS
- FSI_D_CREDIT_LINES
- FSI_D_GUARANTEES
- FSI_D_INVESTMENTS
- FSI_D_LEASES
- FSI_D_LOAN_CONTRACTS
- FSI_D_MERCHANT_CARDS
- FSI_D_MM_CONTRACTS
- FSI_D_MORTGAGES
- FSI_D_SWAPS
- FSI_D_FX_CONTRACTS
- FSI_D_FUTURES

> **Note**
>
> Provide the value for P_FAIL_ON_ERROR parameter as 'N' while executing the DT LOAD_INSTRUMENT_SUMMARY from Batch Maintenance screen.

Copyright © 2011, 2026, Oracle and/or its affiliates.

External Break Process Flow

Breaks Event Data Loaded from External Source Systems

The following approach is recommended when break data is provided by third party source systems and loaded via standard ETL. This is an alternative and possibly a complementary approach to using the FTP Break Identification Process.

Data Flow

- Current period data is moved from staging tables to break funding charges table.
- Prior period data is moved from instrument tables to break funding charges table.

Assumptions: Source systems need to provide information related to breaks in the following fields:

1. **Loans**:
    - Prepayment Indicator – column to indicate if a prepayment has happened in the current period.
    - Closed Indicator – column to indicate if an account is closed in the current period.
    - Ending Principal Balance – column to store the outstanding principal balance.
    - Prepayment Amount – column to store the amount that is pre-paid in case of partial or full pre-payment.
    - Restructured Indicator – column to indicate if the account is restructured.
2. **Term Deposits**:
    - Early Redemption indicator – column to indicate if the account is redeemed early.
    - Redeemed amount – column to indicate that is redeemed in the current period.
    - Closed indicator – column to indicate if the account is closed in the current period.
    - Ending balance – column to store the ending balance of the account.

Copyright © 2011, 2026, Oracle and/or its affiliates.

J

Creating and Executing T2T Processes

The following are the T2T definitions which move data from the staging table(s) to the Break Funding table when Breaks data is available from Source Systems.

- Full Breaks and Partial Breaks:
  - T2T_BROKEN_ACCOUNT_LOANS_FP
  - T2T_BROKEN_ACCOUNT_TD_FP
- Restructuring Accounts:
  - T2T_BROKEN_ACCOUNT_LOANS_R
  - T2T_BROKEN_ACCOUNT_TD_R
- For moving prior period records for Partial and Restructured accounts: The following T2T's will be used for moving prior records into the Break Funding table for Partial Break and Restructuring of accounts.
  - T2T_LOAN_PAR_RESTRUCTURED
  - T2T_MORTGAGES_PAR_RESTRUCTURED
  - T2T_TD_PAR_RESTRUCTURED

For executing these T2T's, follow these steps:

1. Create a batch.
2. Add the relevant tasks to the newly created batch.
3. Add the Transform Data task for updating the Process Run History table.
   The T2T_Process_Update procedure will insert/update the Last Execution Date in the Process Run History table which will be used in the next T2T for fetching prior period AS_OF_DATE.
   - T2T_LOAN_PAR_RESTRUCTURED
   - T2T_MORTGAGES_PAR_RESTRUCTURED
   - T2T_TD_PAR_RESTRUCTURED
4. Execute the new batch process from the Batch Execution screen.

- Load Run IDs for T2Ts
  You can load the data into OFSAA using the OFSAA T2Ts.
- Supported Incremental T2Ts
  The master batch (<INFODOM>_T2T_ALL_INSTRUMENT_TABLES_INCR_LOAD) contains the following incremental T2Ts (for FTP and PFT):

J.1 Load Run IDs for T2Ts

You can load the data into OFSAA using the OFSAA T2Ts.

OFSAA Data Model includes Load Run Identifier as part of the primary key for staging tables. This enables data to be stored for multiple load runs for any date. OFSAA Data Model also has

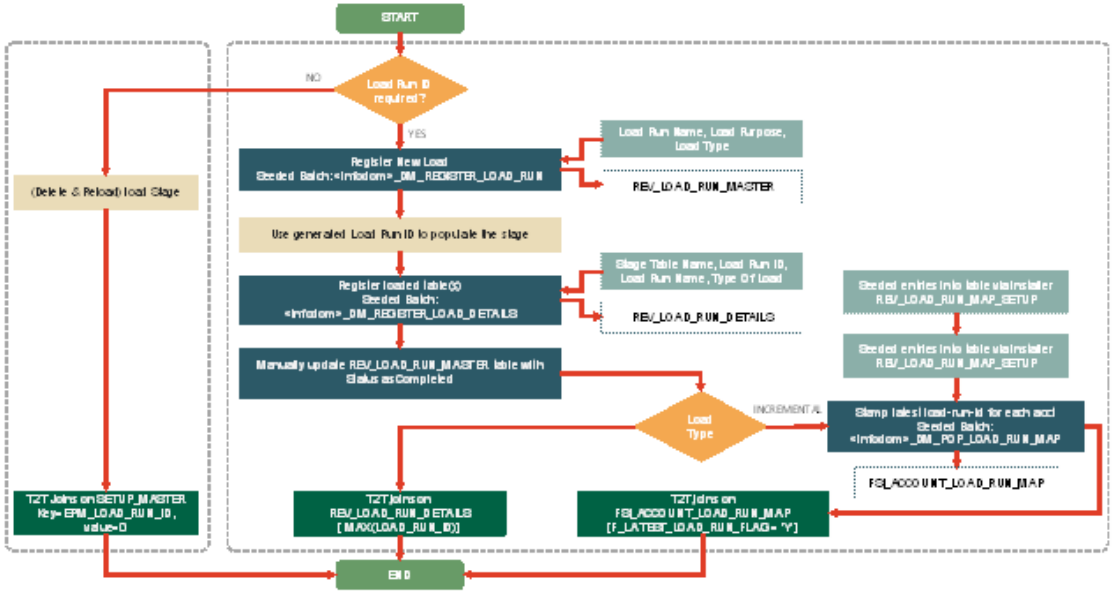
Copyright © 2011, 2026, Oracle and/or its affiliates.

a table to maintain master information about Load Run and can be used for identifying or filtering Load Run during Run execution within OFSAA.

Before loading data into the staging table, generate a Load Run Identifier to stamp the records from the source. These records can be a complete snapshot or can be partial or Incremental Data too. This Load Run Identifier can be generated by calling the function in the OFSAA Atomic Schema named fn_register_load_run (Batch name: DM_REGISTER_LOAD_RUN). The function expects some input parameters (as mentoined in section Incremental T2Ts) and returns a unique Load Run Identifier to the calling program. For OFS FTP and OFS PFT applications, in the SETUP_MASTER_TABLE, the V_COMPONENT_VALUE is seeded with the EPM_LOAD_RUN_ID equals to 0 by default.

The following process flow diagram illustrates the Load Run ID for the T2T process.

Figure J-1 Load Run ID for T2T Process Flow



- Existing T2Ts
  Existing T2Ts are modified to make a join between stage table and SETUP_PAREMATERS_MASTER table on <stage table>.load_run_id = <EPM_LOAD_RUN_ID value> and do the data load to instrument or processing table. Even though changes are made to existing T2Ts, seeded entries in the SETUP_PAREMATERS_MASTER table will make it work without any issues if the Load Run ID column value in the stage table is the default value of zero and not null. If the Load Run ID value in the stage table is any other value other than zero, then customers have to update the EPM_LOAD_RUN_ID parameter value in the SETUP_PAREMATERS_MASTER table with the used Load Run ID value for the load.
- Incremental T2Ts
  Incremental T2Ts are introduced to support the incremental load of data for a particular MIS date.

J.1.1 Existing T2Ts

Existing T2Ts are modified to make a join between stage table and SETUP_PAREMATERS_MASTER table on <stage table>.load_run_id =

Copyright © 2011, 2026, Oracle and/or its affiliates.

<EPM_LOAD_RUN_ID value> and do the data load to instrument or processing table. Even though changes are made to existing T2Ts, seeded entries in the SETUP_PAREMATERS_MASTER table will make it work without any issues if the Load Run ID column value in the stage table is the default value of zero and not null. If the Load Run ID value in the stage table is any other value other than zero, then customers have to update the EPM_LOAD_RUN_ID parameter value in the SETUP_PAREMATERS_MASTER table with the used Load Run ID value for the load.

While using modified existing T2Ts, it has to be used to insert a complete snapshot of the data for a given MIS date. Re-execution for the given MIS date needs deletion of data from processing tables and execution of existing T2Ts. The merging of data is not supported with existing T2T.

With respect to OFS FTP and OFS PFT, the seeded master batch is provided to execute the following T2Ts in a single execution.

If any T2T is missing in the seeded master batch, then the user has to execute manually that T2T batch. This seeded master batch contains the list of T2Ts that are common between OFS FPT and OFS PFT.

> **Note**
>
> This is applicable to OFS FTP. Even though T2Ts are part of the seeded master batch, they will be available as individual T2T batches. You can execute the individual T2T batches alone.

J.1.2 Incremental T2Ts

Incremental T2Ts are introduced to support the incremental load of data for a particular MIS date.

For loading the incremental T2Ts, follow these steps:

1. Verify the REV_LOAD_RUN_MAP_SETUP table for the seeded data.
2. Generate the Load Run ID.
   - **Seeded Batch**: <infodom>_DM_REGISTER_LOAD_RUN
   - **Input**: Load Run Name, Load Purpose, Load Type
   - **Output**: generates Load Run ID in REV_LOAD_RUN_MASTER table
3. Register the Load.
   - **Seeded Batch**:<infodom>_DM_REGISTER_LOAD_DETAILS
   - **Input**: Stage Table Name, Load Run ID, Load Run Name, Type Of Load
   - **Output**: REV_LOAD_RUN_DETAILS
4. Load Data to stage table with generated Load Run ID.
5. Execute the batch to load the data to the FSI_ACCOUNT_LOAD_RUN_MAP table:
   - **Seeded Batch**:<infodom>_DM_POP_LOAD_RUN_MAP
   - **Input**: STAGE_TABLE_NAME, LOAD_RUN_ID, LOAD_RUN_NAME
   - **Output**: FSI_ACCOUNT_LOAD_RUN_MAP
6. Execute the incremental T2T to load the data into the processing table.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- **Input**: FSI_ACCOUNT_LOAD_RUN_MAP, Stage Table
- **Output**: Processing Table

7. Mark the load manually as completed by updating the REV_LOAD_RUN_MASTER table.

J.2 Supported Incremental T2Ts

The master batch (<INFODOM>_T2T_ALL_INSTRUMENT_TABLES_INCR_LOAD) contains the following incremental T2Ts (for FTP and PFT):

- T2T_ANNUITY_CONTRACTS_INCR_LOAD
- T2T_BORROWINGS_INCR_LOAD
- T2T_CASA_INCR_LOAD
- T2T_CREDIT_CARDS_INCR_LOAD
- T2T_GUARANTEES_INCR_LOAD
- T2T_INVESTMENTS_INCR_LOAD
- T2T_LEASES_INCR_LOAD
- T2T_LOAN_CONTRACTS_INCR_LOAD
- T2T_MERCHANT_CARDS_INCR_LOAD
- T2T_MM_CONTRACTS_INCR_LOAD
- T2T_MORTGAGES_INCR_LOAD
- T2T_MUTUAL_FUNDS_INCR_LOAD
- T2T_TERM_DEPOSITS_INCR_LOAD
- T2T_RETIREMENT_ACCOUNTS_INCR_LOAD
- T2T_TRUSTS_INCR_LOAD
- T2T_LOAN_COMMITMENTS_INCR_LOAD
- T2T_FUTURE_CONTRACTS_INCR_LOAD
- T2T_FX_CONTRACTS_PAY_INCR_LOAD
- T2T_FX_CONTRACTS_RCV_INCR_LOAD
- T2T_SWAPS_CONTRACTS_PAY_INCR_LOAD
- T2T_SWAPS_CONTRACTS_RCV_INCR_LOAD
- Executing Incremental T2T
  To load the data from one table to another table (insert new rows in stage table), follow these steps:

J.2.1 Executing Incremental T2T

To load the data from one table to another table (insert new rows in stage table), follow these steps:

1. Navigate to **Common Object Maintenance**, select **Operations**, and then select **Batch Maintenance** to display the Batch Maintenance screen.
2. Select DM_REGISTER_LOAD_RUN seeded batch for Batch ID and click **Add** to display the Task Definition page.

Copyright © 2011, 2026, Oracle and/or its affiliates.

3. Enter a **Task ID** and **Description**.
4. Select **TRANSFORM DATA** for Components.
5. Enter the following details:
   - **Datastore Type**: Leave the default value of “EDW”
   - **Datastore Name**: Leave the default value
   - **IP Address**: Leave the default value
   - **Rule Name**: fn_registerloadrun
   - Parameter List (Load Run Name, Load Purpose, Load Type)
6. Click **Save**.
7. To execute the batch, click the **Batch Execution** option under **Operations**. Select the DM_REGISTER_LOAD_RUN batch and select the corresponding check-box.
8. Click **Execute Batch**. A message is displayed: Batch Triggered Successfully. Click **Ok**.
9. Navigate to Batch Monitor, select the Batch ID click on **Start Monitoring** and verify if the Batch Run is successful.
10. Connect to the database and verify if the entry is made into the REV_LOAD_RUN_MASTER table with a new Load Run ID.
11. From the Batch Maintenance screen, select the seeded batch **DM_REGISTER_LOAD_DETAILS** and click **Add** to open the Task Definition page.
12. For the Parameter List, enter the **Stage Table Name**, **Load Run ID**, **Load Run Name**, and **Type Of Load (I)**.
13. Click **Save**.
14. Navigate to Batch Monitor, select the Batch ID click on **Start Monitoring** and verify if the Batch Run is successful.
15. Verify the REV_LOAD_RUN_DETAILS table for the entry with the latest Load Run ID.
16. Connect to the Database and verify in the REV_LOAD_RUN_MAP table for the seeded data.
17. From the Batch Maintenance screen, select the seeded batch DM_POP_LOAD_RUN_MAP and click **Add** to open the Task Definition page.
18. For the Parameter List, enter the **Stage Table Name**, **Load Run ID**, **Load Run Name**, and **Type Of Load**.

    > **Note**
    >
    > For each State Table, you should add different task and pass the Stage Table parameters.

19. Click **Save**.
20. Navigate to Batch Monitor, select the Batch ID, click **Start Monitoring** and verify if the Batch Run is successful.
21. Connect to the Database and verify if a new entry is made with Load Run ID as 1 in the FSI_ACCOUNT_LOAD_RUN_MAP table.
22. Execute and verify the incremental T2T.

Copyright © 2011, 2026, Oracle and/or its affiliates.

23. Navigate to Batch Monitor, select the Batch ID, click **Start Monitoring** and verify if the Batch Run is successful.
24. Verify the processing table to ensure if the data is inserted.

> **Note**
>
> The following T2Ts are present only in FTP:
>
> - OFSFTPINFO_T2T_FOR_FRA_PAY_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_FRA_RCV_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_OD_ACCOUNTS_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_OPTIONS_CAPFLOOR_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_OPTION_COLLARPAY_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_OPTION_COLLARRCV_INCR_LOAD
> - OFSFTPINFO_T2T_FOR_STG_LC_GUARANTEES_INCR_LOAD
> - OFSFTPINFO_T2T_FX_SWAPS_CONTRACTS_INCR_LOAD

25. Update the REV_LOAD_RUN_MASTER table with Status as Completed. To update, run the following SQL query:

```
UPDATE rev_load_run_master SET LOAD_RUN_STATUS = 'Completed' WHERE Mis_Date =
'DD-MMM-YYYY' AND Load_Run_Name = '<LOAD_RUN_NAME>' and LOAD_RUN_ID = n ;
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

K

Seeded Batches

The following sections describe the seeded batches available in OFS FTP application.

- Load Instrument Summary
- <INFODOM>_PFT_MANAGEMENTLEDGER_LOAD_UNDO
- <INFODOM>_T2T_ALL_INSTRUMENT_TABLES
- <INFODOM>_T2T_ALL_TXN_SUMMARY
- <INFODOM>_T2T_MANAGEMENT_LEDGER_MULTI_SOURCE
- <INFODOM>_T2T_ALL_INS_SUMM_TABLES
- <INFODOM>_DIMENSION_ACCOUNTS
- <INFODOM>_T2T_FOR_FSI_D_CUSTOMER
- <INFODOM>_LOAD_INSTRMNT_SUMMARY
- List of Deprecated Instrument Summary T2Ts
- Load Instrument Summary
  For users who want to consolidate Profitability and Funds Transfer Pricing information from multiple Instrument tables into a single consolidated table, a new process has been created and is described below. The process includes a Data Transformation procedure that will move account level data from each Instrument table into a single table, FSI_D_INST_SUMMARY. The primary use for this new table is that it becomes a single data source for moving Profitability and Funds Transfer Pricing results to FCT tables in the reporting area and replaces the previous process that included separate T2T's for each instrument table. Users may also want to use this table to build Profitability or Funds Transfer Pricing reports directly against process area data.
- <INFODOM>_PFT_MANAGEMENTLEDGER_LOAD_UNDO
  This Batch can be used to UNDO the data loaded into LEDGER CLASS tables.
- <INFODOM>_T2T_ALL_INSTRUMENT_TABLES
  This seeded batch contains the following Instrument Table T2Ts:
- <INFODOM>_T2T_ALL_TXN_SUMMARY
  This seeded batch contains the following Transaction Summary Table T2Ts:
- <INFODOM>_T2T_MANAGEMENT_LEDGER_MULTI_SOURCE
  This seeded batch along with the underlying DTs and T2T enables tracking of data from the Staging table to the Management ledger table using identity codes. For each distinct value of "Data Origin" in the Staging table, a new Identity code is used to populate the Management ledger and FSI_DATA_IDENTITY tables. The value of V_DATA_ORIGIN in the Staging table is mapped to the FSI_DATA_IDENTITY.description column.
- <INFODOM>_T2T_ALL_INS_SUMM_TABLES
  This seeded batch contains the following Instrument Summary Table T2Ts:
- <INFODOM>_DIMENSION_ACCOUNTS
  This seeded batch contains the following SCDs to populate the dim_accounts table:
- <INFODOM>_T2T_FOR_FSI_D_CUSTOMER
  This seeded batch contains the following T2T to load the FSI_D_CUSTOMER table:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- <INFODOM>_LOAD_INSTRMNT_SUMMARY
  This seeded batch contains the following DT to load the FSI_D_INST_SUMMARY from individual INSTRUMENT tables:
- List of Deprecated Instrument Summary T2Ts
  Starting with this Release, the following Instrument Summary T2Ts are deprecated.

K.1 Load Instrument Summary

For users who want to consolidate Profitability and Funds Transfer Pricing information from multiple Instrument tables into a single consolidated table, a new process has been created and is described below. The process includes a Data Transformation procedure that will move account level data from each Instrument table into a single table, FSI_D_INST_SUMMARY. The primary use for this new table is that it becomes a single data source for moving Profitability and Funds Transfer Pricing results to FCT tables in the reporting area and replaces the previous process that included separate T2T's for each instrument table. Users may also want to use this table to build Profitability or Funds Transfer Pricing reports directly against process area data.

In earlier versions, customers who use FTP or Profitability BI applications have been required to run multiple T2Ts'; one for each instrument table to load data into a reporting area table, which was error prone and a time consuming activity.

Additionally, each relevant BI application (e.g. ALM and FTP Analytics, Retail Performance Analytics and Institutional Performance Analytics will provide a single T2T to move data from FSI_D_INST_SUMMARY to FCT_FTP_ACCOUNT_SUMMARY or FCT_PFT_ACCOUNT_SUMMARY as needed to support their specific reporting requirements.

Batch to Load Instrument Summary table:

- Seeded batch name: <INFODOM>_LOAD_INSTRMNT_SUMMARY
- Function name (which gets triggered from the batch): FN_LOAD_INSTRMNT_SUMMARY
- DT name: LOAD_INSTRMNT_SUMMARY
- Description
  This batch is used to load account-level attributes from individual Instrument tables to the FSI_D_INST_SUMMARY table.

K.1.1 Description

This batch is used to load account-level attributes from individual Instrument tables to the FSI_D_INST_SUMMARY table.

The inputs to the batch are (in addition to Batch run ID and FIC MIS date, which are implicit parameters passed to all Batches by the Batch framework):

1. P_FAIL_ON_ERROR: This is an optional parameter. The default value is Y. If the parameter value is Y, then the execution stops at that point where it encounters an error during the execution, and the Batch returns failure status.
   On re-executing the batch with the same input parameter values, the DT skips the tables that have been successfully loaded, and resumes loading the table that failed during last execution.

   If the parameter value is passed as N, then the execution continues even if there is an error, and the Batch returns status as Success. Check Instrmnt_summary_load_status and fsi_message_log tables to see if all Instrument tables are loaded successfully or not. The

Copyright © 2011, 2026, Oracle and/or its affiliates.

instrmnt_summary_load_status.load_status_flag indicates whether the load was S (successful) or F (failed).

On re-executing the batch the execution reloads all tables without skipping the tables that were loaded successfully in the previous execution.

2. P_START_DATE and P_END_DATE: These two are optional parameters. The default value is NULL. If the value for these parameters are passed then the records from Instrument tables will be selected based on these dates. i.e., those records whose as_of_date falls between P_START_DATE and P_END_DATE will be selected (both inclusive).
If these parameter values are NULL, then the records from Instrument tables will be selected based on the input FIC MIS date, i.e., those records whose as_of_date equals the input FIC MIS date.
3. P_TARGET_TABLE: This is an optional parameter. Default value for this parameter is FSI_D_INST_SUMMARY. This is the target Instrument Summary table name which would be loaded from the individual Instrument tables.

> **Note**
>
> Following is the sample request parameters, in case you don't want to use defaults:
> P_FAIL_ON_ERROR => 'N',
>
> P_START_DATE => '20210101',
>
> P_END_DATE => '20210131',
>
> P_TARGET_TABLE => 'FSI_D_INST_SUMMARY'

List of Instrument tables from which the accounts attributes will be loaded are:

- FSI_D_ANNUITY_CONTRACTS
- FSI_D_BORROWINGS
- FSI_D_BREAK_FUNDING_CHARGES
- FSI_D_CASA
- FSI_D_CREDIT_CARDS
- FSI_D_CREDIT_LINES
- FSI_D_FUTURES
- FSI_D_FX_CONTRACTS
- FSI_D_GUARANTEES
- FSI_D_INVESTMENTS
- FSI_D_LEASES
- FSI_D_LOAN_CONTRACTS
- FSI_D_MERCHANT_CARDS
- FSI_D_MM_CONTRACTS
- FSI_D_MORTGAGES
- FSI_D_MUTUAL_FUNDS
- FSI_D_OTHER_SERVICES

Copyright © 2011, 2026, Oracle and/or its affiliates.

- FSI_D_RETIREMENT_ACCOUNTS
- FSI_D_SWAPS
- FSI_D_TERM_DEPOSITS
- FSI_D_TRUSTS

The SETUP_PARAMETERS_MASTER table can be configured with the list of tables to be excluded from the load. For each table name to be excluded, create a row in the SETUP_PARAMETERS_MASTER table with param_name as EXCLUDE_FROM_INSTRMNT_SUMMARY_LOAD, and param_value as the Instrument table name to be excluded. The table names that are excluded from the load will be written to fsi_message_log.

The DT uses the following column UDPs for purpose mentioned in below table. Any user defined column added in the instrument tables must also be added in FSI_D_INST_SUMMARY as well, along with assigning UDP_EXPORT_PFT_OUTPUT.

Please refer DMU guide <section/page number> for additional details regarding the user defined properties.

The following PL/SQL is and example for executing the Load Instrument procedure directly on the schema: declare

```
retVal number;

begin
retVal := FN_LOAD_INSTRMNT_SUMMARY(P_BATCH_RUN_ID   =>
'OFSPFTINFO_LOAD_INSTRMNT_SUMMARY_20210131_1',
P_FIC_MIS_DATE  => '20210131',
P_FAIL_ON_ERROR => 'N',
P_START_DATE    => '20210101',
P_END_DATE      => '20210131',
P_TARGET_TABLE  => 'FSI_D_INST_SUMMARY');
end;
/
```

K.2 <INFODOM>_PFT_MANAGEMENTLEDGER_LOAD_UNDO

This Batch can be used to UNDO the data loaded into LEDGER CLASS tables.

Parameters passed in the batch are given below:

- PFTUNDO.sh,##INFODOM##_<Batch Run Id> <Identity Code>-<As of Date> L where:
  - INFODOM: Information Domain name.
  - BATCH_RUN_ID: Any unique identifier to identify the batch execution.
  - Identity Code: Identity code generated by ledger load that needs to be removed.
  - As of Date: MIS DATE of Ledger Load in MM/DD/YYYY format.

K.3 <INFODOM>_T2T_ALL_INSTRUMENT_TABLES

This seeded batch contains the following Instrument Table T2Ts:

- T2T_ANNUITY_CONTRACTS

Copyright © 2011, 2026, Oracle and/or its affiliates.

- T2T_BORROWINGS
- T2T_CASA
- T2T_CREDIT_CARDS
- T2T_GUARANTEES
- T2T_INVESTMENTS
- T2T_LEASES
- T2T_LOAN_CONTRACTS
- T2T_MERCHANT_CARDS
- T2T_MM_CONTRACTS
- T2T_MORTGAGES
- T2T_MUTUAL_FUNDS
- T2T_OTHER_SERVICES
- T2T_RETIREMENT_ACCOUNTS
- T2T_TERM_DEPOSITS
- T2T_TRUSTS
- T2T_LOAN_COMMITMENTS

You can edit this batch to add or remove other Instrument Table T2Ts.

K.4 <INFODOM>_T2T_ALL_TXN_SUMMARY

This seeded batch contains the following Transaction Summary Table T2Ts:

- T2T_ANNUITY_TXNS_SUMMARY
- T2T_BORROWINGS_TXNS_SUMMARY
- T2T_CASA_TXNS_SUMMARY
- T2T_CREDIT_CARDS_TXNS_SUMMARY
- T2T_CREDIT_LINES_TXNS_SUMMARY
- T2T_GUARANTEES_TXNS_SUMMARY
- T2T_INVESTMENTS_TXNS_SUMMARY
- T2T_LEASES_TXNS_SUMMARY
- T2T_LOAN_CONTRACTS_TXNS_SUMMARY
- T2T_MERCHANT_CARDS_TXNS_SUMMARY
- T2T_MM_CONTRACTS_TXNS_SUMMARY
- T2T_MORTGAGES_TXNS_SUMMARY
- T2T_MUTUAL_FUNDS_TXNS_SUMMARY
- T2T_OTHER_SERVICES_TXNS_SUMMARY
- T2T_RETIREMENT_ACCTS_TXNS_SUMMARY
- T2T_TERM_DEPOSITS_TXNS_SUMMARY
- T2T_TRUSTS_TXNS_SUMMARY

Copyright © 2011, 2026, Oracle and/or its affiliates.

You can edit this batch to add or remove other Transaction Summary Table T2Ts.

K.5 <INFODOM>_T2T_MANAGEMENT_LEDGER_MULTI_SOURCE

This seeded batch along with the underlying DTs and T2T enables tracking of data from the Staging table to the Management ledger table using identity codes. For each distinct value of "Data Origin" in the Staging table, a new Identity code is used to populate the Management ledger and FSI_DATA_IDENTITY tables. The value of V_DATA_ORIGIN in the Staging table is mapped to the FSI_DATA_IDENTITY.description column.

The seeded batch named <INFODOM>_T2T_MANAGEMENT_LEDGER_MULTI_SOURCE will have the following three tasks in the sequence:

1. A DT named FN_MGMT_LED_LOAD_TEMP_TABLE, which will populate the T2T_FSI_D_MANAGEMENT_LEDGER table with unique identity_code values for each of the distinct v_data_origin values in the STG_GL_DATA table. It accepts two parameters - Target Table Name and Staging Table Name.
2. T2T named T2T_MANAGEMENT_LEDGER_MULTI_SOURCE, which will use the identity codes from the table T2T_FSI_D_MANAGEMENT_LEDGER for loading data from STG_GL_DATA to FSI_D_MANAGEMENT_LEDGER.
3. Another DT named FN_MGMT_LED_LOAD_DATA_IDENTITY, which will populate the FSI_DATA_IDENTITY table with the information about the management ledger load. It maps the V_DATA_ORIGIN column in the STG_GL_DATA table to the DESCRIPTION column in the FSI_DATA_IDENTITY table. It will also delete the records from the T2T_FSI_D_MANAGEMENT_LEDGER table.

Note

1. The results of the DT execution is written to the FSI_MESSAGE_LOG table.
2. The OOB T2T uses STG_GL_DATA as the staging table. The T2T has is customized to use any other staging table as the source.
3. If a custom Staging table is used instead of STG_GL_DATA, the table needs to have the columns FIC_MIS_DATE and V_DATA_ORIGIN.

K.6 <INFODOM>_T2T_ALL_INS_SUMM_TABLES

This seeded batch contains the following Instrument Summary Table T2Ts:

- T2T_INS_SUMM_TERM_DEPOSITS
- T2T_INS_SUMM_RETIREMENT_ACC
- T2T_INS_SUMM_OTHER_SERVICES
- T2T_INS_SUMM_MUTUAL_FUNDS
- T2T_INS_SUMM_MM_CONTRACTS
- T2T_INS_SUMM_MERCHANT_CARDS
- T2T_INS_SUMM_LOANS
- T2T_INS_SUMM_LEASES

Copyright © 2011, 2026, Oracle and/or its affiliates.

- T2T_INS_SUMM_LDGR_STAT_INST
- T2T_INS_SUMM_INVESTMENTS
- T2T_INS_SUMM_GUARANTEES
- T2T_INS_SUMM_CREDIT_LINES
- T2T_INS_SUMM_CREDIT_CARDS
- T2T_INS_SUMM_CASA
- T2T_INS_SUMM_BREAK_FUND_CHG
- T2T_INS_SUMM_BORROWINGS
- T2T_INS_SUMM_ANNUITY
- T2T_INS_SUMM_TRUSTS

K.7 <INFODOM>_DIMENSION_ACCOUNTS

This seeded batch contains the following SCDs to populate the dim_accounts table:

- STG_ANNUITY_CONTRACTS
- STG_BILLS_CONTRACTS
- STG_BORROWINGS
- STG_CARDS
- STG_CASA
- STG_COMMITMENT_CONTRACTS
- STG_FUTURES
- STG_FX_CONTRACTS
- STG_GUARANTEES
- STG_INVESTMENTS
- STG_LC_CONTRACTS
- STG_LEASES_CONTRACTS
- STG_LOAN_CONTRACTS
- STG_MM_CONTRACTS
- STG_MUTUAL_FUNDS
- STG_OD_ACCOUNTS
- STG_OPTION_CONTRACTS
- STG_REPO_CONTRACTS
- STG_RETIREMENT_ACCOUNTS
- STG_TD_CONTRACTS
- STG_TRUSTS
- STG_MERCHANT_CARDS
- STG_CREDIT_DERIVATIVES

Copyright © 2011, 2026, Oracle and/or its affiliates.

> **Note**
>
> The DIMENSION_ACCOUNTS seeded batch is common across all the EPM applications and hence the STG_CREDIT_DERIVATIVES SCD is part of the seeded batches.
> However, this task does not apply to OFS PFT because this staging table is not available in the PFT Data Model.

K.8 <INFODOM>_T2T_FOR_FSI_D_CUSTOMER

This seeded batch contains the following T2T to load the FSI_D_CUSTOMER table:

- T2T_FSI_D_CUSTOMER

K.9 <INFODOM>_LOAD_INSTRMNT_SUMMARY

This seeded batch contains the following DT to load the FSI_D_INST_SUMMARY from individual INSTRUMENT tables:

- FSI_D_MUTUAL_FUNDS
- FSI_D_OTHER_SERVICES
- FSI_D_RETIREMENT_ACCOUNTS
- FSI_D_TERM_DEPOSITS
- FSI_D_TRUSTS
- FSI_D_ANNUITY_CONTRACTS
- FSI_D_BORROWINGS
- FSI_D_CASA
- FSI_D_CREDIT_CARDS
- FSI_D_CREDIT_LINES
- FSI_D_GUARANTEES
- FSI_D_INVESTMENTS
- FSI_D_LEASES
- FSI_D_LOAN_CONTRACTS
- FSI_D_MERCHANT_CARDS
- FSI_D_MM_CONTRACTS
- FSI_D_MORTGAGES
- FSI_D_SWAPS
- FSI_D_FX_CONTRACTS
- FSI_D_FUTURES

K.10 List of Deprecated Instrument Summary T2Ts

Starting with this Release, the following Instrument Summary T2Ts are deprecated.

Copyright © 2011, 2026, Oracle and/or its affiliates.

OFSAA recommends to use the LOAD_INSTRMNT_SUMMARY DT to move data to the FSI_D_INST_SUMMARY table.

- T2T_INS_SUMM_ANNUITY
- T2T_INS_SUMM_BORROWINGS
- T2T_INS_SUMM_BREAK_FUND_CHG
- T2T_INS_SUMM_CASA
- T2T_INS_SUMM_CREDIT_CARDS
- T2T_INS_SUMM_CREDIT_LINES
- T2T_INS_SUMM_GUARANTEES
- T2T_INS_SUMM_INVESTMENTS
- T2T_INS_SUMM_LEASES
- T2T_INS_SUMM_LOANS
- T2T_INS_SUMM_MERCHANT_CARDS
- T2T_INS_SUMM_MM_CONTRACTS
- T2T_INS_SUMM_MORTGAGES
- T2T_INS_SUMM_MUTUAL_FUNDS
- T2T_INS_SUMM_OTHER_SERVICES
- T2T_INS_SUMM_RETIREMENT_ACC
- T2T_INS_SUMM_TERM_DEPOSITS
- T2T_INS_SUMM_TRUSTS

Copyright © 2011, 2026, Oracle and/or its affiliates.

L

FTP Web Services

FTP web services (v8.0) provide the capability to query Transfer Pricing results from existing account records and execute Transfer Pricing calculations against ad-hoc instruments in near real-time. The FTP web service calculations utilize the same transfer pricing methodologies and calculations that are used when running standard transfer pricing processes. FTP web services apply to all FTP calculations including Transfer Rates (all TP Methods), TP Add-on rates (all methods), and Breakage Charges. FTP web services are available to be called by any third party system within the bank.

FTP web services support the following use cases:

- Ability to query FTP information for existing instrument records for a specified date.
- Ability to query FTP Daily Rate Card information for standard products.
- Ability to execute an existing TP Process in real-time, which can refer to any TP Rule, Prepayment Rule, Adjustment Rule, and (optionally) an Alternate Rate Output Mapping assumption for a single ad-hoc instrument record.
- Ability to calculate an Economic Loss Breakage Charge in real-time, for an ad-hoc instrument record.

Topics:

- Important Changes in OFSAA Release 8.1
- How to test and execute the FTP Web Services?
- Configure the FTP Web Services
- Use the Web API Testing Tool
- Description of FTP Web Services
- Web Service name: getTPRate
- Web Service name: getEconomicCostWithParameters
- Web Service name: getPricing
- Web Service name: getPricingData
- Payment Schedule Enhanced FTP Web Service
- Web Service Changes for GDPR
- Java Programs for Testing the FTP Web Service
- Additional Configuration Needed for Economic Cost Web Services

- Important Changes in OFSAA Release 8.1
  Starting with release 8.1, FTP is supporting only Restful services for FTP Web Services. The previous versions were supporting SOAP-based services. OFSAA has decided to move to REST as it is becoming the de facto communication mechanism across various platforms. REST is faster and uses less bandwidth. It is also easier to integrate REST with the existing applications with no need to re-factor infrastructure. This enables integration faster.

Copyright © 2011, 2026, Oracle and/or its affiliates.

- How to test and execute the FTP Web Services?
  All the web services require a JSON template (request.json) to be submitted along with the web service request. The JSON template provides the necessary details to the web service allowing it to perform the requested action.
- Configure the FTP Web Services
  FTP Web services configuration parameters are given in the ofsaaws.properties file under $FIC_WEB_HOME/WEB-INF/classes directory.
- Use the Web API Testing Tool
  Testing of these web services can be done by using any standard web API testing tool available on the Internet like Postman, HTTP Master, or Rest-assured.
- Description of FTP Web Services
  The following use cases refer to a single java program (FTPWSclient.jar), which is available from the OFSAA Support Team.
- Web Service Name: getTPRate
  Ability to query existing instrument records that contain FTP information for a specified date.
- Web Service Name: getEconomicCostWithParameters
  Ability to calculate a breakage charge in real-time, for a specified instrument record.
- Web Service name: getPricing
  Ability to execute an existing TP Process which can refer to any TP Rule, Prepayment Rule, Adjustment Rule, and (optionally) an Alternate Rate Output Mapping assumption in real time for a single ad-hoc instrument record.
- Web Service name: getPricingData
  The getPricing service invokes the server process in real-time and waits for the calculated values to be returned by the engine. If the server processing time exceeds the allotted time limit, the web service will return only the REQUEST_ID. In this case, you can re-query the results using the getPricingData web service.
- Payment Schedule Enhanced FTP Web Service
  Users have the option to pass payment schedule details with the Economic Cost web service if required.
- Web Service Changes for GDPR
  A new parameter is introduced for FTP `getEconomicCostWithparameters` and `getTPRate` web services that determine the lookup column for account number Field in FSI instrument tables.
- Java Programs for Testing the FTP Web Services
  The following Java programs that call the FTP web services and are referenced in the above examples:
- Additional Configuration Needed for Economic Cost Web Services
  Economic Cost Web Services make Java Native Interface call to invoke the calculation engine. So the expectation is that the calculation engine libraries to be available in the same box, where the web service is deployed. In your environment, if the webs server and FIC Server are on different boxes, follow these additional setups to use the Economic Cost Web Services:

L.1 Important Changes in OFSAA Release 8.1

Starting with release 8.1, FTP is supporting only Restful services for FTP Web Services. The previous versions were supporting SOAP-based services. OFSAA has decided to move to REST as it is becoming the de facto communication mechanism across various platforms.

Copyright © 2011, 2026, Oracle and/or its affiliates.

REST is faster and uses less bandwidth. It is also easier to integrate REST with the existing applications with no need to re-factor infrastructure. This enables integration faster.

L.2 How to test and execute the FTP Web Services?

All the web services require a JSON template (request.json) to be submitted along with the web service request. The JSON template provides the necessary details to the web service allowing it to perform the requested action.

General steps for testing/executing the web service calls:

1. Testing of these web services can be done by using any standard web API testing tool like a postman, HTTP Master, or Rest-assured.
2. We have provided examples of testing these services using postman. This is detailed under Section 'Using web API testing tool'. For other web API testing tools, the process will be similar.
3. To support the testing of the FTP web services, we provide a sample java program that can be called directly. This jar file allows you to call the FTP web services. Please ensure the FTPWSclient.jar file is in the same folder as the JSON file(s). To receive a copy of the sample program, please raise a service request with the OFSAA support team. You can also view sample java code at the end of this document. Usage of this test program is detailed under the section Use a Java Client Program.

L.3 Configure the FTP Web Services

FTP Web services configuration parameters are given in the ofsaaws.properties file under $FIC_WEB_HOME/WEB-INF/classes directory.

The following parameters must be provided:

```
infodom= e.g. OFSPFTINFO
userId= e.g. ftpuser1
pollFrequency=1000 [in milliseconds]
pollIterations=3
Instrument_tables=210,230~FSI_D_TERM_DEPOSITS;130~FSI_D_MORTGAGES;110,120,180~
FSI_D_LOAN_CONTRACTS;160~FSI_D_CREDIT_LINES;
cpp_component= services
```

For a full description of how to install and configure FTP web services, see the *OFS Funds Transfer Pricing Web Service Configuration* section in the OFS Analytical Applications Infrastructure Installation and Configuration Guide.

L.4 Use the Web API Testing Tool

Testing of these web services can be done by using any standard web API testing tool available on the Internet like Postman, HTTP Master, or Rest-assured.

Download and install the web API testing tool (example Postman) from its distribution website.

Steps for testing using a web API testing tool are detailed for all the web service methods along with the required configuration, input parameters, and expected response. For more information, see Description of FTP Web Services.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Use a Java Client Program

To use a java client program for web API testing, follow these steps:

1. Create the required JSON and populate the appropriate parameters. Save it in a file (Request.json)
2. Open Command Line and navigate to the location of the JSON files.
3. Run the following command mentioning the relevant web service and the JSON file. Example: `java -jar FTPWSclient.jar <parameter> Request.json <RestURL>`
4. The results will appear on the command line.

L.5 Description of FTP Web Services

The following use cases refer to a single java program (FTPWSclient.jar), which is available from the OFSAA Support Team.

If required, please raise a service request to receive a copy of the file. It is assumed that anyone looking to deploy the FTP web services will have some basic knowledge of web service development and will be constructing similar wrapper programs for their specific use cases. This java program can be used for testing purposes.

L.6 Web Service Name: getTPRate

Ability to query existing instrument records that contain FTP information for a specified date.

A relationship manager may want to retrieve the FTP information for existing account records from the database. For example, a customer may have multiple accounts with the bank, one is a personal loan, and another is a term deposit. The relationship manager can query the transfer rate information for these existing accounts and use the actual margin information (positive or negative) as the basis to offer rates for the new products. Providing this kind of information in real-time allows the banker to make informed pricing decisions.

Rest URL format:

```
http://<ServerName>:port/<contextName>/services/OFSAAFTPService/getTPRate
```

The attributes given with angular brackets must be replaced with the appropriate value in your environment.

- **ServerName**: This is the name of the host where these services are deployed.
- **port**: This is the port number on the server where it is listening for an incoming request.
- **contextname**: This is the name of the web archive for the FTP Economic Cost web service (FTPWebServiceAXIS).

Input JSON format:

The following table provides an example of the required inputs in JSON format:

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table L-1 An example of the required inputs in JSON format

Function Name	JSON FORMAT

getTPRate	{
	"ACCOUNT_NUMBER": "TLN1TLN1",
	"AS_OF_DATE": "20140920",
	"INSTRUMENT_TYPE_CODE":"230"
	}

Function parameters (supplied within request.JSON file):

- Account Number
- As-of-Date
- Instrument Type Code
- Test this service using the Java command line Test Program
  To test the service using the java command line test program, follow these steps:
- Test using Web API Testing Tool (example Postman)
  To test using the Web API Testing Tool (example Postman), follow these steps:

L.6.1 Test this service using the Java command line Test Program

To test the service using the java command line test program, follow these steps:

1. Create the required JSON and populate the appropriate parameters. Save it in a file (Request.JSON)
2. Open Command Line and navigate to the location of the JSON files.
3. Run the command mentioning the relevant web service and the JSON file.

Example: `java -jar FTPWSclient.jar Request.json <RestURL>`

The results will appear on the command line.

This screenshot illustrates how the web service call is made and what the return information looks like when running from the command line.

Example: `java -jar FTPWSClient.jar Request.json <Restful_URL>`

> **Note**
>
> "TP" in the above command indicates you are calling the getTPRate program for this web service. A value of "EC" would indicate you are calling the getEconomicCost program.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure L-1 Putty Snapshot

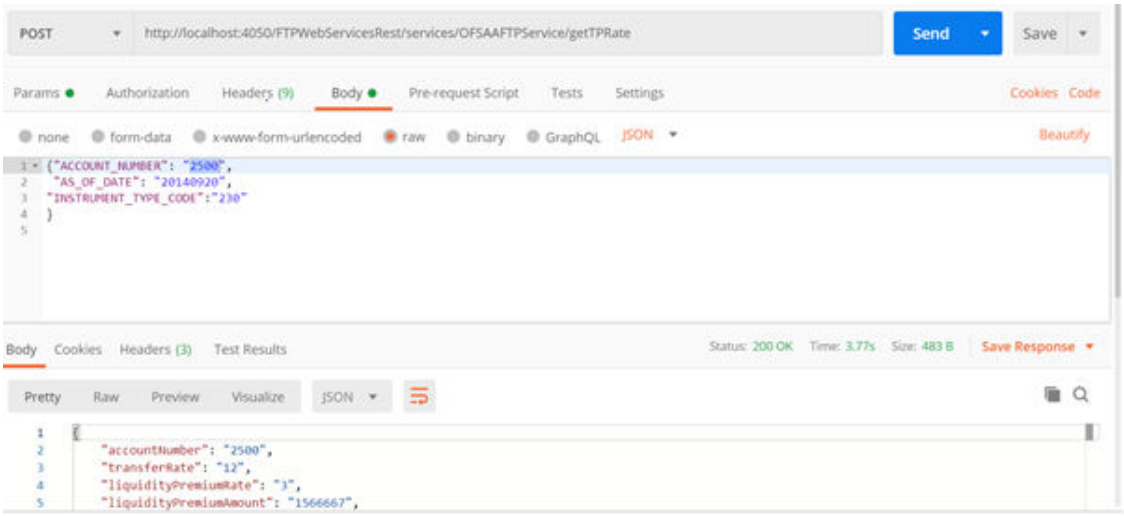


L.6.2 Test using Web API Testing Tool (example Postman)

To test using the Web API Testing Tool (example Postman), follow these steps:

1. Create a new request from the tool's New Menu.
2. Select the request type as POST.
3. Enter the FTP Rest services URL.
4. In the headers section set the 'Content-Type= application/JSON'.
5. We have detailed the request JSON format for all the web services.
6. Create the JSON in the required format.
7. Copy the JSON and paste it in the Body's raw section of the tool.
8. Press the send button.
9. The request is submitted to the requested web service.
10. The service is invoked on the server and the requested/calculated values are shown in the response section of the tool.
11. The following image shows the above steps:

Figure L-2 Web API Testing Tool



Copyright © 2011, 2026, Oracle and/or its affiliates.

The getTPRate web service works with data from the following tables. The specific table where the instrument record lookup will execute is identified in the request.JSON based on INSTRUMENT_TYPE_CODE.

- FSI_D_TERM_DEPOSITS (230)
- FSI_D_MORTGAGES (130)
- FSI_D_LOAN_CONTRACTS (180)
- FSI_D_CREDIT_LINES (160)

This list of tables can be extended by including additional tables in the ofsaaws.properties file.

Return values:

- TRANSFER_RATE
- LIQUIDITY_PREMIUM_AMT
- LIQUIDITY_PREMIUM_RATE
- BASIS_RISK_COST_AMT
- BASIS_RISK_COST_RATE
- PRICING_INCENTIVE_AMT
- PRICING_INCENTIVE_RATE
- OTHER_ADJUSTMENTS_AMT
- OTHER_ADJUSTMENTS_RATE
- ALL_IN_TP_RATE

L.7 Web Service Name: getEconomicCostWithParameters

Ability to calculate a breakage charge in real-time, for a specified instrument record.

This web service supports the calculation of an economic loss breakage charge. This calculation can be embedded in a front-line application to provide breakage charge quotes to customers. This calculation supports all three types of breaks including, Full break, Partial break, and Change in attributes break. A few examples of common break events are as follows:

- A customer wants to redeem their term deposit before expiry. The bank charges a fee equal to the actual economic loss incurred on the account due to the early redemption of the deposit.
- A customer approaches the bank to prepay the total outstanding balance on their Mortgage loan and want to know what the penalty will be. The bank can calculate the economic breakage charge to be paid by the customer and advise the customer before the payoff.
- A customer approaches the bank wanting to convert the interest rate on their loan from fixed to floating or vice versa. The bank can calculate the economic loss (or gain) and advise the customer on any fees they would incur.
- The number format is enhanced so that the breakage charge web service returns the economic cost with full precision.

Rest URL format:

Copyright © 2011, 2026, Oracle and/or its affiliates.

`http://<ServerName>:port/<contextName>/services/OFSAAFTPService/getEconomicCostWithParameters` The attributes given with angular brackets should be replaced with the appropriate value in your environment.

- The server name is the name of the host where these services are deployed.
- Port: This is the port number on the server where it is listening for the incoming request.
- Context Name is the name of the web archive for the FTP Economic Cost web service (FTPWebServiceAXIS).
- Input JSON format
  The following table provides an example of the required inputs in JSON format:
- Test the Service using the Java Command-line Test Program
  To test this service using the Java command-line test program, follow these steps:
- Test using Web API Testing tool (example Postman)
  To test this service using the Web API Testing tool, follow these steps:

L.7.1 Input JSON format

The following table provides an example of the required inputs in JSON format:

Function Name: `getEconomicCostWithParameters`

JSON Format:

```
{"CURRENT":
{
"ACCOUNT_NUMBER": "TLN1TLN1",
"AS_OF_DATE": "20140920",
"CURRENT_NET_RATE": "8",
"ORIGINAL_TERM_TO_MATURITY": "12",
"ORIGINAL_TERM_MULTIPLIER": "M",
"PAYMENT_FREQUENCY": "1",
"PAYMENT_FREQUENCY_MULTIPLIER": "M",
"ADJUSTABLE_TYPE_CODE": "0",
"CURRENT_GROSS_BOOK_BALANCE": "48000",
"ACCRUAL_BASIS_CODE": "3",
"AMORTIZATION_TERM": "12",
"AMORTIZATION_TERM_MULTIPLIER": "M",
"AMORTIZATION_TYPE_CODE": "100",
"CURRENT_GROSS_PAR_BALANCE": "48000",
"CURRENT_PAYMENT_AMOUNT": "6959",
"INTEREST_RATE_CODE": "7777",
"LAST_REPRICING_DATE": "20140701",
"MATURITY_DATE": "20150701",
"NEXT_PAYMENT_DATE": "20141001",
"NEXT_REPRICING_DATE": "20150701",
"ORIGINAL_GROSS_PAR_BALANCE": "80000",
"ORIGINATION_DATE": "20140701",
"ISSUE_DATE": "20140701",
"REMAINING_NO_OF_PAYMENTS": "10",
"LAST_PAYMENT_DATE": "20140901",
"INTEREST_TYPE_CODE": "1",
"COMPOUNDING_BASIS_CODE": "120",
"REPRICING_FREQUENCY": "0",
"REPRICING_FREQUENCY_MULTIPLIER": "M",
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
"TP_RATE": "5.844472",
"INSTRUMENT_TYPE_CODE": "",
"TP_METHOD_CODE": "4"
},
"PRIOR": {
"ACCOUNT_NUMBER": "TLN1TLN1",
"AS_OF_DATE": "20140920",
"CURRENT_NET_RATE": "8",
"ORIGINAL_TERM_TO_MATURITY": "12",
"ORIGINAL_TERM_MULTIPLIER": "M",
"PAYMENT_FREQUENCY": "1",
"PAYMENT_FREQUENCY_MULTIPLIER": "M",
"ADJUSTABLE_TYPE_CODE": "0",
"CURRENT_GROSS_BOOK_BALANCE": "68000",
"ACCRUAL_BASIS_CODE": "3",
"AMORTIZATION_TERM": "12",
"AMORTIZATION_TERM_MULTIPLIER": "M",
"AMORTIZATION_TYPE_CODE": "100",
"CURRENT_GROSS_PAR_BALANCE": "68000",
PAYMENT_AMOUNT": "6959",
_RATE_CODE": "7777",
RICING_DATE": "20140701",
_DATE": "20150701",
MENT_DATE": "20141001",
RICING_DATE": "20150701",
_GROSS_PAR_BALANCE": "80000",
ION_DATE": "20140701",
TE": "20140701",
G_NO_OF_PAYMENTS": "10",
MENT_DATE": "20140901",
_TYPE_CODE": "1",
ING_BASIS_CODE": "120",
"REPRICING_FREQUENCY": "0",
"REPRICING_FREQUENCY_MULTIPLIER": "M",
"TP_RATE": "5.844472",
"INSTRUMENT_TYPE_CODE": "",
"TP_METHOD_CODE": "4"
},
"MINIMUM_CHARGE": "-999999",
"INTEREST_ONLY_FLAG": "0",
"RATE_SPREAD": "0",
"HOLIDAY_CALENDAR": "0",
"INTERPOLATION_METHOD": "1",
"ASSET_LIABILITY": "1",
"SERVICE_TYPE": "B",
"EXCLUDE_ACCRUED_INTEREST": "N",
"PAYMENTSCHEDULE": {
"SCHEDULE": [
{
"AMOUNT": "10000",
"DATE": "20140715"
},
{
"AMOUNT": "20000",
"DATE": "20140815"
},
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
{
"AMOUNT": "25000",
"DATE": "20140915"
},
{
"AMOUNT": "30000",
"DATE": "20141015"
},
{
"AMOUNT": "35000",
"DATE": "20141115"
}
]
}
}
```


Function parameters: The following attributes are passed for both the current period (post break) record and prior period (before break) record via request.json:

- ACCOUNT_NUMBER
- AS_OF_DATE
- CURRENT_NET_RATE
- ORIGINAL_TERM_TO_MATURITY
- ORIGINAL_TERM_MULTIPLIER
- PAYMENT_FREQUENCY
- PAYMENT_FREQUENCY_MULTIPLIER
- ADJUSTABLE_TYPE_CODE
- CURRENT_GROSS_BOOK_BALANCE
- ACCRUAL_BASIS_CODE
- AMORTIZATION_TERM
- AMORTIZATION_TERM_MULTIPLIER
- AMORTIZATION_TYPE_CODE
- CURRENT_GROSS_PAR_BALANCE
- CURRENT_PAYMENT_AMOUNT
- INTEREST_RATE_CODE (The specified IRC will be used both for Transfer Pricing the record if TP Rate is not given and for looking up current rates to calculate the Breakage Charge) • LAST_REPRICING_DATE
- MATURITY_DATE
- NEXT_PAYMENT_DATE
- NEXT_REPRICING_DATE
- ORIGINAL_GROSS_PAR_BALANCE
- ORIGINATION_DATE
- ISSUE_DATE
- REMAINING_NO_OF_PAYMENTS

Copyright © 2011, 2026, Oracle and/or its affiliates.

- LAST_PAYMENT_DATE
- INTEREST_TYPE_CODE
- COMPOUNDING_BASIS_CODE
- REPRICING_FREQUENCY
- REPRICING_FREQUENCY_MULTIPLIER
- TP_RATE (If provided, the TP Method choice below, is not applicable. This TP rate will be used instead of re-calculating. If TP Rate is not given, then it will be a lookup from a database or calculated using the following method):
  - INSTRUMENT_TYPE_CODE
  - TP_METHOD_CODE (10 - Cash Flow: Zero Discount Factor or 4 - Straight term)
  - INTERPOLATION_METHOD (1- Linear, 2- Cubic spline, 3- Quartic spline)
  - ASSET_LIABILITY (1- Asset, 2- Liability)
  - SERVICE_TYPE (A- Actual or S-Simulated) If "A" is selected (recommended) details will be logged to an audit table, FSI_ECONOMIC_COST_HIST
  - EXCLUDE_ACCRUED_INTEREST (Y - Yes, N - No)

To calculate the Breakage Charge, a Transfer Rate must be available. The transfer rate can be arrived in the following ways:

- If the TP_RATE is provided in the request.JSON it will be used.
- If TP_RATE is not provided, then ACCOUNT_NUMBER, AS_OF_DATE, and INSTRUMENT_TYPE_CD, if provided, will be used to lookup the Transfer Rate from the existing instrument record.
- If neither of the above cases is provided, then the Transfer Rate will be calculated using the indicated TP Method and Interest Rate Curve.

The getEconomicCostWithParameters web service works with data from the following tables. The specific table where the instrument record lookup will execute is identified in the request.json based on INSTRUMENT_TYPE_CODE.

- FSI_D_TERM_DEPOSITS (230)
- FSI_D_MORTGAGES (130)
- FSI_D_LOAN_CONTRACTS (180)
- FSI_D_CREDIT_LINES (160)

This list of tables can be extended by including additional tables in the ofsaaws.properties file.

L.7.2 Test the Service using the Java Command-line Test Program

To test this service using the Java command-line test program, follow these steps:

1. Create the required JSON and populate the appropriate parameters. Save it in a file (Request.json)
2. Open Command Line and navigate to the location of the JSON files.
3. Run the command mentioning the relevant web service and the JSON file.
   Example: `java -jar FTPWSClient.jar Request.json <RESTFUL_URL>`

Copyright © 2011, 2026, Oracle and/or its affiliates.

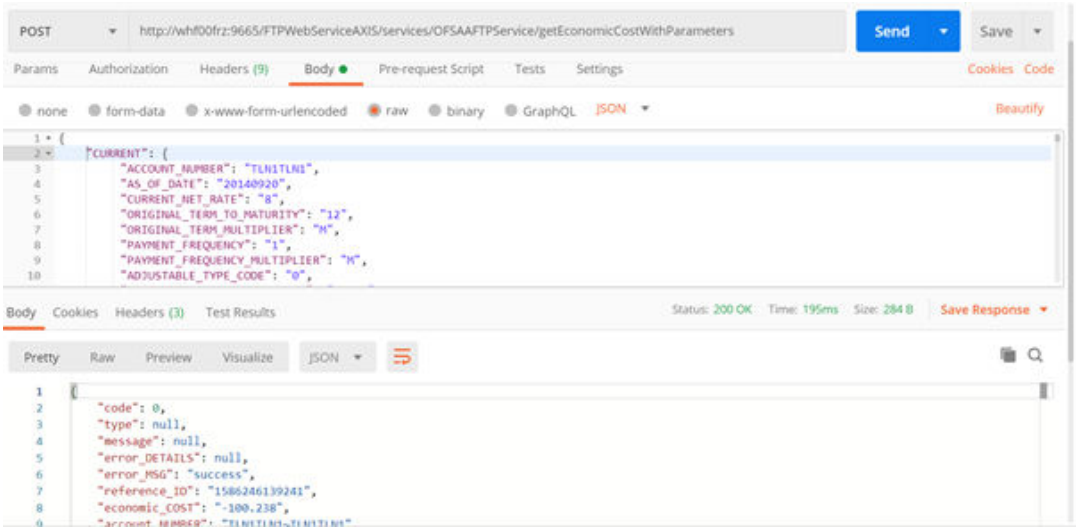
> **Note**
>
> "EC" in the above command indicates you are calling the getEconomicCostWithParameters program for this web service. A value of "TP" would indicate you are calling the getTPRate program.

L.7.3 Test using Web API Testing tool (example Postman)

To test this service using the Web API Testing tool, follow these steps:

1. Create a new request from the tool's New Menu.
2. Select the request type as POST.
3. Enter the FTP Rest services URL.
4. In the headers section set the 'Content-Type= application/JSON'.
5. We have detailed the request JSON format for all the web services.
6. Create the JSON in the required format.
7. Copy the JSON and paste it in the Body -> raw section of the tool.
8. Press the send button.
9. The request is submitted to the requested web service.
10. The service is invoked on the server and the requested/calculated values are shown in the response section of the tool.
    The following screenshot illustrates how the web service call is made using the tool.

Figure L-3 Web API Testing Tool (example Postman)



L.8 Web Service name: getPricing

Ability to execute an existing TP Process which can refer to any TP Rule, Prepayment Rule, Adjustment Rule, and (optionally) an Alternate Rate Output Mapping assumption in real time for a single ad-hoc instrument record.

Copyright © 2011, 2026, Oracle and/or its affiliates.

This web service supports provisioning of FTP rate quotes for more complicated instrument records, which are not supported by the Daily FTP Rate Card process. For cases where pricing for non-standard products is required, details for the entire data record can be passed through this web service to the TP Engine, and results will be generated in nearly real-time. This web service works in the following way:

- The data record provided in JSON is inserted by the java web service program getPricing into the FSI_PM_GENERATED_INSTRMTS table.
- The Process ID of the TP Process (references TP Rule and Adjustment Rule) given in JSON is executed to generate transfer pricing results for the row of data.
- The Product ID given in the JSON is used to lookup the TP Method and Adjustment assumptions.
- If processing is completed within the time set in ofsaaws.properties file, then request ID and results will be returned automatically.
- If processing is not completed within the time set in the ofsaaws.properties, then only the request ID is returned.
- In case the results are not returned due to longer processing time, users can set up a second web service call to re-query the results using the getPricingData web service program bypassing the request ID that was returned by the getPricing call.

Rest URL format: `http://<ServerName>:port/<contextName>/rest-api/ftpPcdRestservice/ v1/getPricing`

The attributes given with angular brackets should be replaced with the appropriate value in your environment.

- **ServerName**: This is the name of the host where these services are deployed.
- **Port**: This is the port number on the server where it is listening for an incoming request.
- **contextName**: This is the name of the web archive for the FTP Application (OFSAA). Input JSON format:

The following table provides an example of the required inputs in JSON format.

getPricing:

```
{    "processId" :"510291",
    "instrumentParameters" : [{
        "accountNumber" : "NEW_ACCOUNT_1",
        "asOfDate" : "20181031",
        "currentNetRate" : "12",
        "originalTermToMaturity" : "5",
        "originalTermMultiplier" : "Y",
        "paymentFrequency" : "1",
        "paymentFrequencyMultiplier" : "M",
        "adjustableTypeCode" : "0",
        "currentGrosBookBalance" : "250000",
        "accuralBasisCode" : "2",
        "amortizationTerm" : "5",
        "amortizationTermMultiplier" : "Y",
        "amortizationTypeCode" : "100",
        "currentGrossParBalance" : "250000",
        "currentPaymentAmount" : "4167",
        "interestRateCode" : "9999",
        "lastRepricingDate" : "20181031",
        "maturityDate" : "20230711",
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
        "nextPaymentDate" : "20180812",
        "nextRepricingDate" : "20230711",
        "originalGrossparBanalce" : "250000",
        "originationDate" : "20180503",
        "issueDate" : "20181031",
        "remainingNoOfPayments" : "0",
        "lastPaymentDate" : "20181031",
        "interestTypeCode" : "2",
        "compoundingBasisCode" : "160",
        "repricingFrequency" : "0",
        "repricingFrequencyMultiplier" : "M",
        "productCode" : "1013",
        "currency" : "USD",
        "orgPaymentAmt" : "4167"
    }]
}
```


Response:


```
{
    "status": "success",
    "data": {
        "requestId": 61,
            "calculatedData": [
    {
        "requestId": "61",
        "accountNumber": "TestingProdFee",
        "transferRate": "7",
        "liquidityPremiumRate": "1",
        "pricingIncentiveRate": "9",
        "basisRiskCostRate": "9",
        "otherAdjustmentRate": "9",
        "allInTPRate": "28",
        "asOfDate": "2018-11-10 00:00:00.0"
    }
    ]
    },
    "messages": null
}
```


Function parameters (for getPricing):

- PROCESS_ID
- ACCOUNT_NUMBER
- AS_OF_DATE
- CURRENT_NET_RATE
- ORIGINAL_TERM_TO_MATURITY
- ORIGINAL_TERM_MULTIPLIER
- PAYMENT_FREQUENCY
- PAYMENT_FREQUENCY_MULTIPLIER
- ADJUSTABLE_TYPE_CODE

Copyright © 2011, 2026, Oracle and/or its affiliates.

- CURRENT_GROSS_BOOK_BALANCE
- ACCRUAL_BASIS_CODE
- AMORTIZATION_TERM
- AMORTIZATION_TERM_MULTIPLIER
- AMORTIZATION_TYPE_CODE
- CURRENT_GROSS_PAR_BALANCE
- CURRENT_PAYMENT_AMOUNT
- ORG_PAYMENT_AMT
- INTEREST_RATE_CODE
- LAST_REPRICING_DATE
- MATURITY_DATE
- NEXT_PAYMENT_DATE
- NEXT_REPRICING_DATE
- ORIGINAL_GROSS_PAR_BALANCE
- ORIGINATION_DATE
- ISSUE_DATE
- REMAINING_NO_OF_PAYMENTS
- LAST_PAYMENT_DATE
- INTEREST_TYPE_CODE
- COMPOUNDING_BASIS_CODE
- REPRICING_FREQUENCY
- REPRICING_FREQUENCY_MULTIPLIER
- PRODUCT_ID
- ISO_CURRENCY_CD

Return values:

- REQUEST_ID
- ACCOUNT_NUMBER
- TRANSFER_RATE
- LIQUIDITY_PREMIUM_RATE
- BASIS_RISK_COST_RATE
- PRICING_INCENTIVE_RATE
- OTHER_ADJUSTMENTS_RATE
- ALL_IN_TP_RATE
- AS_OF_DATE

If the TP Process exceeds the allotted time limit, the web service will return only the REQUEST_ID. In this case, you can re-query the results using the getPricingData web service.

- Test this service using the Java Command-line test program
  To test this service using the Java command-line test program, follow these steps:

Copyright © 2011, 2026, Oracle and/or its affiliates.

- Test using Web API Testing tool (example Postman)
  To test this service using the Web API Testing tool, follow these steps:

L.8.1 Test this service using the Java Command-line test program

To test this service using the Java command-line test program, follow these steps:

1. Create the required JSON and populate the appropriate parameters. Save it in a file (Request.JSON).
2. Open Command Line and navigate to the location of the JSON files.
3. Run the command mentioning the relevant web service and the JSON file.
   Example: `java -jar PricingClient.jar request.json <RESTFUL_URL> userId password`

> **Note**
>
> This is a secured service and needs a valid FTP UserID and password to be passed in the command-line. The passed user id must have the FTP Process Execution function assigned with the mapped role.

L.8.2 Test using Web API Testing tool (example Postman)

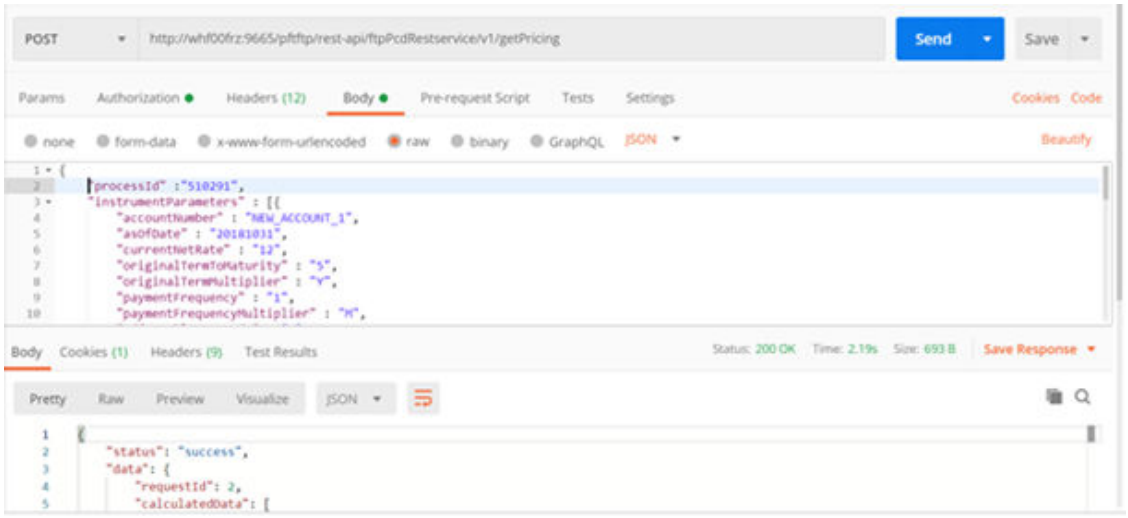
To test this service using the Web API Testing tool, follow these steps:

1. Create a new request from the tool's New Menu.
2. Select the request type as POST.
3. Enter the FTP Rest services URL.
4. In the headers section set the 'Content-Type= application/JSON'.
5. We have detailed the request JSON format for all the web services.
6. Create the JSON in the required format.
7. Copy the JSON and paste it in the Body 'raw section of the tool.
8. Press the send button.
9. The request is submitted to the requested web service.
10. The service is invoked on the server and the requested/calculated values are shown in the response section of the tool.
11. This is a secured web service and uses basic authorization. So authorization section of the tool should be populated with a valid user Id and password of the FTP application.
12. The header section of the tool needs additional parameters 'userID' with the value of a valid FTP User ID.

The following screenshot illustrates how the web service call is made using the tool.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure L-4 Web API Testing Tool (example Postman)



L.9 Web Service name: getPricingData

The getPricing service invokes the server process in real-time and waits for the calculated values to be returned by the engine. If the server processing time exceeds the allotted time limit, the web service will return only the REQUEST_ID. In this case, you can re-query the results using the getPricingData web service.

Rest URL format: `http://<ServerName>:port/<contextName>/rest-api/ftpPcdRestservice/ v1/getPricingData`

The attributes given with angular brackets should be replaced with the appropriate value in your environment.

- ServerName: This is the name of the host where these services are deployed.
- Port: This is the port number on the server where it is listening for an incoming request.
- contextName: This is the name of the web archive for the FTP Application (OFSAA).

Input Format:

```
Function Name: getPricingData
XML Format:
{
REQUEST_ID: 8
}
```

Function parameters (for getPricingData):

- REQUEST_ID
- Return values:
  - REQUEST_ID
  - ACCOUNT_NUMBER
  - TRANSFER_RATE
  - LIQUIDITY_PREMIUM_RATE

Copyright © 2011, 2026, Oracle and/or its affiliates.

  - BASIS_RISK_COST_RATE
  - PRICING_INCENTIVE_RATE
  - OTHER_ADJUSTMENTS_RATE
  - ALL_IN_TP_RATE
  - AS_OF_DATE
-
-
- Test this service using the Java Command-line test program
  To test this service using the Java command-line test program, follow these steps:
- Test using Web API Testing tool (example Postman)
  To test this service using the Web API Testing tool, follow these steps:

L.9.1 Test this service using the Java Command-line test program

To test this service using the Java command-line test program, follow these steps:

1. Create the required JSON and populate the appropriate parameters. Save it in a file (Request.json).
2. Open Command Line and navigate to the location of the JSON files.
3. Run the command mentioning the relevant web service and the JSON file.
   Example: `java -jar PricingClient.jar input1.json <RESTFUL_URL>`

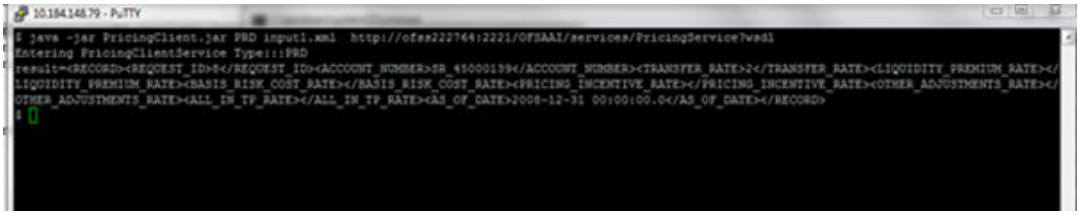
L.9.2 Test using Web API Testing tool (example Postman)

To test this service using the Web API Testing tool, follow these steps:

1. Create a new request from the tool's New Menu.
2. Select the request type as GET.
3. Enter the FTP Rest services URL.
4. In the headers section set the 'Content-Type= text/html'.
5. In param tabs of the request add key as 'requestId' and value as the request data for which I want to retrieve data. Example requestId=8.
6. Press the send button.
7. The request is submitted to the requested web service.
8. The service is invoked on the server and the requested/calculated values are shown in the response section of the tool.
9. This is a secured web service and uses basic authorization. So, the authorization section of the tool should be populated with a valid user Id and password of the FTP application.
10. The header section of the tool needs additional parameters 'userID' with the value of a valid FTP application User ID.
    This screenshot illustrates how the web service call is made using the tool.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Figure L-5 Web API Testing Tool (example Postman)



Related Topics:

- See the OFS Funds Transfer Pricing Applications Pack Installation and Configuration Guide for more information on installing and configuring FTP Web Services.

L.10 Payment Schedule Enhanced FTP Web Service

Users have the option to pass payment schedule details with the Economic Cost web service if required.

Backward compatibility is still provided for old payment schedule JSON tags, but the same Payment Schedule value is passed for both prior and current records and the engine will not fail.

The following is the Eco_Cost_with_2_Payment_Schedule_tags.json file. While using this web service, note the following points:

- You need to provide the Payment Schedule for both prior and current periods, else the engine will not process.
- Backward compatibility is still provided for the old payment schedule JSON tags, but the same Payment Schedule value will be passed for both prior and current records and the engine will not fail.
- The amortization code needs to be 800, 801, and 802 to ensure that the engine uses the provided payment schedule. Supported TP methods are only the Straight Term and Cash Flow: Zero Discount Factor.
- Even if the TP method is Straight Term, the Payment Schedule will be used by Breakage Charge calculation.

Example JSON for payment Schedule

Function Name: `getEconomicCostWithParameters`

JSON Format:

```
{
    "CURRENT":{
        "ACCOUNT_NUMBER":"TLN1TLN1",
        "AS_OF_DATE":"20140920",
        "CURRENT_NET_RATE":"8",
        "ORIGINAL_TERM_TO_MATURITY":"12",
        "ORIGINAL_TERM_MULTIPLIER":"M",
        "PAYMENT_FREQUENCY":"1",
        "PAYMENT_FREQUENCY_MULTIPLIER":"M",
        "ADJUSTABLE_TYPE_CODE":"0",
        "CURRENT_GROSS_BOOK_BALANCE":"48000",
        "ACCRUAL_BASIS_CODE":"3",
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
        "AMORTIZATION_TERM":"12",
        "AMORTIZATION_TERM_MULTIPLIER":"M",
        "AMORTIZATION_TYPE_CODE":"800",
        "CURRENT_GROSS_PAR_BALANCE":"48000",
        "CURRENT_PAYMENT_AMOUNT":"6959",
        "INTEREST_RATE_CODE":"7777",
        "LAST_REPRICING_DATE":"20140701",
        "MATURITY_DATE":"20150701",
        "NEXT_PAYMENT_DATE":"20141001",
        "NEXT_REPRICING_DATE":"20140701",
        "ORIGINAL_GROSS_PAR_BALANCE":"80000",
        "ORIGINATION_DATE":"20140701",
        "ISSUE_DATE":"20140701",
        "REMAINING_NO_OF_PAYMENTS":"10",
        "LAST_PAYMENT_DATE":"20140901",
        "INTEREST_TYPE_CODE":"1",
        "COMPOUNDING_BASIS_CODE":"120",
        "REPRICING_FREQUENCY":"0",
        "REPRICING_FREQUENCY_MULTIPLIER":"M",
        "TP_RATE":"5.844472",
        "INSTRUMENT_TYPE_CODE":"",
        "TP_METHOD_CODE":"10",
        "PAYMENTSCHEDULE":{
            "SCHEDULE":[
            {
                "AMOUNT":"10000",
                "DATE":"20140715"
            },
            {
                "AMOUNT":"20000",
                "DATE":"20140815"
            },
            {
                "AMOUNT":"25000",
                "DATE":"20140915"
            },
            {
                "AMOUNT":"30000",
                "DATE":"20141015"
            },
            {
                "AMOUNT":"35000",
                "DATE":"20141115"
            }
            ]
        }
    },
    "PRIOR":{
                "ACCOUNT_NUMBER":"TLN1TLN1",
                "AS_OF_DATE":"20140920",
                "CURRENT_NET_RATE":"8",
                "ORIGINAL_TERM_TO_MATURITY":"12",
                "ORIGINAL_TERM_MULTIPLIER":"M",
                "PAYMENT_FREQUENCY":"1",
                "PAYMENT_FREQUENCY_MULTIPLIER":"M",
                "ADJUSTABLE_TYPE_CODE":"0",
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
                "CURRENT_GROSS_BOOK_BALANCE":"68000",
                "ACCRUAL_BASIS_CODE":"3",
                "AMORTIZATION_TERM":"12",
                "AMORTIZATION_TERM_MULTIPLIER":"M",
                "AMORTIZATION_TYPE_CODE":"800",
                "CURRENT_GROSS_PAR_BALANCE":"68000",
                "CURRENT_PAYMENT_AMOUNT":"6959",
                "INTEREST_RATE_CODE":"7777",
                "LAST_REPRICING_DATE":"20140701",
                "MATURITY_DATE":"20150701",
                "NEXT_PAYMENT_DATE":"20141001",
                "NEXT_REPRICING_DATE":"20150701",
                "ORIGINAL_GROSS_PAR_BALANCE":"80000",
                "ORIGINATION_DATE":"20140701",
                "ISSUE_DATE":"20140701",
                "REMAINING_NO_OF_PAYMENTS":"10",
                "LAST_PAYMENT_DATE":"20140901",
                "INTEREST_TYPE_CODE":"1",
                "COMPOUNDING_BASIS_CODE":"120",
                "REPRICING_FREQUENCY":"0",
                "REPRICING_FREQUENCY_MULTIPLIER":"M",
                "TP_RATE":"5.844472",
                "INSTRUMENT_TYPE_CODE":"",
                "TP_METHOD_CODE":"10",
                "PAYMENTSCHEDULE":{
                "SCHEDULE":[
            {
                "AMOUNT":"10000",
                "DATE":"20140715"
            },
            {
                "AMOUNT":"20000",
                "DATE":"20140815"
            },
            {
                "AMOUNT":"25000",
                "DATE":"20140915"
            },
            {
                "AMOUNT":"30000",
                "DATE":"20141015"
            },
            {
                "AMOUNT":"35000",
                "DATE":"20141115"
            }
            ]
            }
        },
            "MINIMUM_CHARGE":"-999999",
            "INTEREST_ONLY_FLAG":"0",
            "RATE_SPREAD":"0",
            "HOLIDAY_CALENDAR":"0",
            "INTERPOLATION_METHOD":"1",
            "ASSET_LIABILITY":"1",
            "SERVICE_TYPE":"B",
```


Copyright © 2011, 2026, Oracle and/or its affiliates.

```
                "EXCLUDE_ACCRUED_INTEREST":"N"
}
```

L.11 Web Service Changes for GDPR

A new parameter is introduced for FTP `getEconomicCostWithparameters` and `getTPRate` web services that determine the lookup column for account number Field in FSI instrument tables.

Properties Filename: `ofsaaws.propertries`

ParameterName: `accountColumnName`

Accepted Value: `Actual_Account_Number` or `Account_Number`. (Default is `Account_Number`)

Both these columns are present in FSI instrument tables. This value is required to be set up in the `ofsaaws.properties` file present in the WEB-INF folder in your web service deployment location before calling the web service.

Based on your data redaction policy, use the appropriate column configuration.

L.12 Java Programs for Testing the FTP Web Services

The following Java programs that call the FTP web services and are referenced in the above examples:

- EcoCostClient.java
- PricingClient.java

L.13 Additional Configuration Needed for Economic Cost Web Services

Economic Cost Web Services make Java Native Interface call to invoke the calculation engine. So the expectation is that the calculation engine libraries to be available in the same box, where the web service is deployed. In your environment, if the webs server and FIC Server are on different boxes, follow these additional setups to use the Economic Cost Web Services:

1. Copy the $FIC_DB_HOME /bin/ftp to a folder in Webserver. Update the LD_LIBRARY_PATH pointing to this folder.
2. Update the user profile on the web server box with the following:
   - Update LD_LIBRARY_PATH to point to the Oracle client and Java installations.
   - Add the following in the NLSPATH=$FIC_DB_HOME/conf/en_US/ofsmsg.cat file.

```
export NLSPATH
INIPATH=$FIC_DB_HOME/conf
export INIPATH
export LDR_CNTRL=MAXDATA=0x40000000
```

Upgrade of Economic Cost web service from v8.0.6.x to v8.1

To upgrade of Economic cost web service from version 8.0.6 to 8.1 requires the deletion of few legacy jars from the deployment area. The script ofs_ftp_webservice_clean_up.sh deletes

Copyright © 2011, 2026, Oracle and/or its affiliates.

these legacy jars. Copy this script file on the OFSAA server and execute it. You must execute the script before the creation of the web service war file.

Copyright © 2011, 2026, Oracle and/or its affiliates.

M
Glossary

This glossary lists the popular terminologies from the current document.

Table M-1 Glossary of Terms

Term	Description
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.
Absolute Repricing Pattern	A type of repricing pattern commonly used for instruments with date dependent repricing characteristics. See also: User-Defined Repricing Patterns
Account Table	Stores a detailed set of transaction-level data attributes of 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 is used for defining the Spread from Interest Rate Code and Redemption Curve transfer pricing methods. You can choose the origination date, last repricing date, or the last day of the associated calendar period as the assignment date.
Assumptions	A set of values or methodologies that apply to dimension values.
Attributed Dimension	A dimension whose members can have other properties or qualifiers known as dimension attributes.
Calculation Mode	Any of the two ways, standard or remaining term, of calculating transfer rates and options costs supported by Oracle Funds Transfer Pricing.
Cash Flow Edits	A process that allows you to verify and correct your Instrument table data.
Cash Flow Edit Logic	A set of checks that are performed in a specified order on the Instrument table data during a Cash Flow Edits Process run.
Cash Flow Transfer Pricing Methods	Methods that generate transfer rates based on the cash flow characteristics of the instruments. These methods are 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 the calculation of interest.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
Data Set	A dimension is 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.
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 non-hierarchical.
Dimension Attribute	A property or qualifier that further describes a dimension member. An attribute can be anything such As-of-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 numeric string that uniquely identifies each member of a dimension. Dimension identifiers are non-translatable, as they are the same regardless of the language context. Each dimension has its own unique set of columns in the Analytical Applications Infrastructure staging tables that serve as the dimension identifier for that dimension.
Dimension Member	The values are used to populate dimension columns in the 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 the 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 custom prepayment model 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.
Entered Currency	The currency in which business transactions take place. Entered currency might be different from functional Currency. See also: Functional currency
Fact Table	A table that contains data uniquely differentiated by dimension columns.
Filter	A business object that filters the source data that is used as input to a process.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
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.
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 the Moving Averages transfer pricing method.
Hierarchy	A structure of dimension members organized by parent-child relationships.
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 the Moving Averages transfer pricing method.
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 the 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.
Lag Term	Indicates that the interest rate must be referenced from a yield curve for a date earlier than the assignment date. The 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 the lowest level dimension numbers. This is the dimension member that is mapped to the instrument data.
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, adjustments, or option costs, obtained from transfer pricing and option cost calculations 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 at the City level, while Geography members such as Texas and Oklahoma belong at the State level. The designation of the level is the same across all hierarchies within a dimension. In other words, Texas is always a state in all Geography hierarchies.
Lookup Method	The method used to calculate prepayment rates for prepayment driver values that do not fall on the nodes defined for that particular prepayment driver.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
Low Discrepancy Sequences	Sequences, also known as quasirandom sequences, are 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	FSI_D_MANAGEMENT_LEDGER, the most central fact table in the OFS Analytical Applications Infrastructure. It contains a ledger and some statistical data and highly aggregated information, such as cash and other assets, and equity. This table supports Oracle Financial Services analytical applications.
Mid-period Repricing Option	Allows you to take into 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 non-cash flow transfer pricing methods.
Node	A dimension value is located anywhere in a hierarchy.
Node Level Assumption	An assumption is assigned to a dimension value at a level higher than a leaf level. A node-level assumption is associated with a business rule that uses a hierarchical dimension. See also: Product dimension
Non-cash 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 timeline and amount of a specific payment in a payment pattern.
Payment Pattern	A user-defined custom amortization pattern. Payment patterns allow 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.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
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 is used to manage the association of prepayment methodologies and rates to various product-currency combinations.
Process Data	Data required to produce results.
Processing Table	Instrument table available for, or included in, a Transfer Pricing Process rule run.
Product Dimension	A Product dimension on which Oracle Funds Transfer Pricing product level assumptions are based. The Product dimension should be populated with your product chart of account at a level of detail appropriate for assigning transfer pricing assumptions to your data.
Propagation Pattern	Allows you to specify parameters, such as source tables, used in the 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.
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 is used to generate transfer rates in the Spread from Interest Rate and Spread from Note Rate methods.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
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. The reference currency is also the currency with which an Interest Rate Code is associated. When you create an Interest Rate Code, you select the reference currency.
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
Relative Repricing Pattern	A type of repricing pattern comprising a series of repricing events driven by user-defined timelines. A relative repricing pattern is used for instruments where the repricing is determined by the elapsed time since origination. See also: repricing pattern
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 calendar period end 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 Funds 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 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	The method used to interpolate rates on the valuation curve for terms that fall between given points. Also, known as the Interpolation Method.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Table M-1 (Cont.) Glossary of Terms

Term	Description
Split Payment Pattern	A split pattern contains multiple sets of payment patterns under a single amortization code. See also: payment pattern
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 Funds Transfer Pricing.
Stochastic Rate Index Rule	A business rule is used to establish a relationship between an Interest Rate Code, a risk-free yield curve, and other Interest Rate Codes. Stochastic Rate Index rules are used to generate forward rates for option cost calculations using stochastic interest rate models.
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 non-amortizing instruments.
Transfer Pricing Rule	A business rule is used to manage the association of transfer pricing methodologies and certain parameters used in option costing to various product-currency combinations.
Transfer Pricing Process	A business rule is 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 OFS Analytical Applications Infrastructure. AAI supports user-defined dimensions containing hierarchies and attributes, as well as user-defined simple dimensions.
User-defined Payment Pattern	See: payment pattern
User-defined Repricing Pattern	See: repricing pattern
Yield Curve Term	The point on the yield curve that the system references to calculate transfer rates.

Copyright © 2011, 2026, Oracle and/or its affiliates.

Glossary

Copyright © 2011, 2026, Oracle and/or its affiliates.

Index

Copyright © 2011, 2026, Oracle and/or its affiliates.