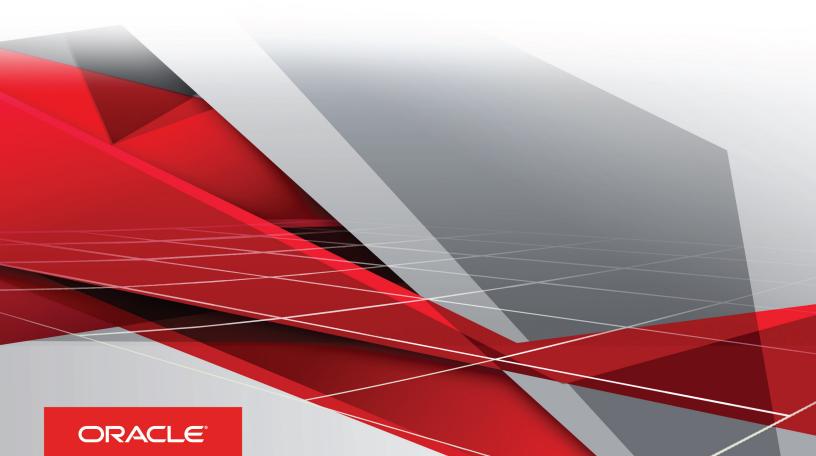
Oracle Fusion

Accounting Hub Implementing Oracle Fusion Accounting Hub

Release 12



Oracle® Fusion Accounting Hub Implementing Oracle Fusion Accounting Hub

Part Number E73429-05

Copyright © 2011-2018, Oracle and/or its affiliates. All rights reserved.

Authors: Kathryn Wohnoutka, Barbara Kostelec, Barbara Snyder, Asra Alim, Essan Ni Jirman, P. S. G. V. Sekhar, Michael Laverty, Sanjay Mall, Angie Shahi

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 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 installed on the hardware, and/ or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. 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 and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon 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, Opteron, the AMD logo, and the AMD Opteron 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.

The business names used in this documentation are fictitious, and are not intended to identify any real companies currently or previously in existence.

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 customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Contents

Pretace	
Oracle Fusion Accounting Hub Overview	1
Define Fusion Accounting Hub Configuration for Rapid Implementation: Overview	_
Creating an Oracle Fusion Accounting Hub Rapid Implementation Project: Worked Example	2
Oracle Fusion Accounting Hub Features: Overview	2
Oracle Fusion Accounting Hub: How It Works	3
Accounting Configuration Offerings: Overview	7
Implementation Options for Oracle Fusion Accounting Hub: Overview	7
Setup Data Import and Export for Oracle Fusion Accounting Hub: Explained	10
Manage Application Implementation	15
Define Enterprise Structures for Fusion Accounting Hub Enterprise Structures: Overview	19
·	19 2-
Enterprise Structures Business Process Model: Explained Global Enterprise Configuration: Points to Consider	23
Modeling Your Enterprise Management Structure in Oracle Fusion: Example	24
Running Diagnostic Tests for Enterprise Structures Setup Data: Explained	
Define Common Financials Configuration for Rapid Implementation	28
Define Initial Configuration with the Enterprise Structures Configurator	
	28 28 37
FACs for Define Initial Configuration	28 37
FAQs for Define Initial Configuration Define Enterprise for Fusion Accounting Hub	28 37 40
Define Enterprise for Fusion Accounting Hub	28 37 40 41
Define Enterprise for Fusion Accounting Hub Define Legal Jurisdictions and Authorities	
Define Enterprise for Fusion Accounting Hub	28 37 40 41 42



3 Define Financial Reporting Structures	53
Representing Your Enterprise Structure in Your Financial Reporting Structure: Overview	53
Financial Enterprise Structure: How It Fits Together	53
Modeling Your Financial Reporting Structure in Oracle Fusion: Example	56
Manage Chart of Accounts	59
FAQs for Manage Charts of Accounts	71
Manage Value Sets	71
Manage Account Hierarchies	76
Manage Accounting Combinations	97
Maintain Segment Value Attributes	98
FAQs for Maintain Segment Values Attributes	105
Deploy Flexfields	106
Manage Cross-Validation Rules	117
Manage Chart of Accounts Mapping	123
FAQ for Manage Chart of Accounts Mapping	124
Manage DRM Synchronization	125
Manage Chart of Accounts Synchronization wtih DRM	125
Manage Accounting Hierarchies with DRM	127
FAQs for Manage DRM Synchronization	131
Define Calendars	131
FAQs for Define Calendars	134
Manage Currencies	135
FAQs for Manage Currencies	136
Manage Conversion Rate Types	136
FAQs for Manage Conversion Rate Types	139
Manage Daily Rates	140



	145
Ledgers and Subledgers: Explained	145
Ledgers: Points to Consider	145
Financial Ledgers: How They Fit Together	148
Creating Primary Ledgers: Example	151
Specify Ledger Options	152
FAQs for Specify Ledger Options	154
Review and Submit Accounting Configuration	155
FAQs for Review and Submit Accounting Configuration	175
Assign Access to Ledger	175
Manage General Ledger Security	176
FAQs for Manage General Ledger Security	199
Manage Reporting Currencies	200
Manage Ledger Sets	202
Manage Accounting and Reporting Sequences	203
FAQs for Manage Accounting and Reporting Sequences	208
Manage Journal Approval Rules	208
Manage AutoPost Criteria Sets	21-
FAQs for Manage AutoPost Criteria Sets	213
-	213 213
FAQs for Manage AutoPost Criteria Sets	
FAQs for Manage AutoPost Criteria Sets	
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets	213
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components	213 217
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close	213 217 217
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close	213 217 217 220
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries	213 217 217 220 221
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries	213 217 217 220 221 226
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries	213 217 217 220 221 226 232
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries FAQs for Manage Allocations and Periodic Entries	213 217 217 220 221 226 232 238
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries Define Allocation Rules	213 217 220 221 226 232 238 238
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries Define Allocation Rules FAQs for Define Allocation Rules	213 217 220 221 226 232 238 239 249
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries Define Allocation Rules FAQs for Define Allocation Rules Define Allocation Rule Sets	213 217 217 220 221 220 232 238 238 249 250
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries Define Allocation Rules FAQs for Define Allocation Rules Define Allocation Rule Sets FAQs for Define Allocation Rule Sets	217 217 220 221 226 232 238 238 249 250 258
FAQs for Manage AutoPost Criteria Sets Manage Journal Reversal Criteria Sets Define Period Close Components Manage Period Close FAQs for Manage Period Close Manage Allocations and Periodic Entries Manage Recurring Entries Manage Allocation Entries FAQs for Manage Allocations and Periodic Entries Define Allocation Rules FAQs for Define Allocation Rules Define Allocation Rule Sets FAQs for Define Allocation Rule Sets Working with Point of View and Allocation Components	217 217 220 221 226 232 238 239 249 250 258



6	Define Applications Coexistence Configuration	277
	Integration with Oracle E-Business Suite and Oracle PeopleSoft: Overview	277
	FAQs for Define Applications Coexistence Configuration	278
	Define Applications Coexistence Configuration for E-Business Suite	278
	FAQs for Integration with Oracle E-Business Suite	280
	Define Applications Integration with PeopleSoft	282
	FAQs for Define Applications Integration with PeopleSoft	317
7	Define Accounting Transformation	319
	Accounting Transformation Configuration: Overview	319
	Analyze Accounting Events	329
	Register Source Systems	338
	Implement Accounting Event Interfaces	352
	Overview of Event APIs	354
	Details of Event API	356
	Create Accounting Integration	372
	FAQs for Create Account Integration	378
	Manage Accounting Rules	378
	Migrate the Configuration	396
	Secure Accounting Transformations	397
	Create and Process Subledger Journal Entries	398
8	Manage Subledger Accounting	401
	Manage Subledger Accounting: Overview	401
	Create Accounting	401
	Create Subledger Journal Adjustment	409
	Review Subledger Journal Entry	412
	Subledger Balances	414
	Advanced Features	417
	Manage Subledger Accounting Reporting	431



9	Define and Maintain Intercompany Processing Rules	437
	Manage Intercompany System Options	437
	Manage Intercompany Organizations	438
	Define Invoicing Options	439
	Manage Intercompany Balancing Rules	440
	Manage Secondary and Clearing Company Balancing Options	448
	Manage Intercompany Transactions	454
	FAQs for Manage Intercompany Transactions	464
	Reconcile Accounts	464
10	Define Consolidations	467
	Consolidation Method: Overview	467
	Reporting Only Consolidation Method: Explained	467
	Balance Transfer Consolidation Method: Explained	469
	Reporting Only Versus Balance Transfer: Points to Consider	471
	Balance Transfers: Overview	471
	Performing Consolidations: Examples	472
	Reporting Consolidation with Multiple Levels: Examples	473
	Preparing Eliminations: Examples	475
	Financial Management Integration Option: Overview	480
	Mapping Segments to Financial Management Dimensions: Explained	481
	Configure ERP Integrator: Overview	482
	FAQs for Define Consolidations	483
11	Define Hyperion Financial Management Integration	485
	Oracle Hyperion Financial Management: Overview	485
	Integration with Hyperion Financial Management: Overview	486
12	Define Budget Configuration	487
	Budget Uploads: Overview	487
	Importing Budget Data from a Flat File: Explained	488
	Loading Data to the Budget Interface Table: Explained	490
	Importing Budget Data from a Spreadsheet: Explained	491
	Budget Correction with Oracle ADF Desktop Integrator: Explained	493
	Oracle Hyperion Planning: Overview	494
	Integration with Oracle Enterprise Planning and Budgeting Cloud Service: Overview	495



13	Define Financial Reporting	497
	Financial Reporting Center Implementation: Points to Consider	497
	Financial Reporting Center: How It Works	497
	Setting Up Your Financial Reporting Center: Critical Choices	500
	Migrating Financial Reports: Instructions	503
	Financial Reporting Permissions in BI Catalog: Example	504
	Oracle Fusion General Ledger Predefined Reports	505
	General Ledger Subject Areas, Folders, and Attributes: Explained	508
	FAQs for Define Financial Reporting	511
	Video Tutorials to Define a Basic Financial Report	511
14	Define Extensions for Fusion Accounting Hub	513
	Define Custom Enterprise Scheduler Jobs for Fusion Accounting Hub	513



Preface

This preface introduces information sources that can help you use the application.

Oracle Applications Help

Use the help icon (?) to access Oracle Applications Help in the application. If you don't see any help icons on your page, click the Show Help icon (?) in the global header. Not all pages have help icons. You can also access Oracle Applications Help at https://fusionhelp.oracle.com.

Using Applications Help

Watch: This video tutorial shows you how to find help and use help features.

Additional Resources

- Community: Use Oracle Applications Customer Connect to get information from experts at Oracle, the partner community, and other users.
- Guides and Videos: Go to the Oracle Help Center to find guides and videos.
- Training: Take courses on Oracle Cloud from Oracle University.

Documentation Accessibility

For information about Oracle's commitment to accessibility, see the Oracle Accessibility Program .

Comments and Suggestions

Please give us feedback about Oracle Applications Help and guides! You can send e-mail to: oracle_fusion_applications_help_ww_grp@oracle.com.





Oracle Fusion Accounting Hub Overview

Define Fusion Accounting Hub Configuration for Rapid Implementation: Overview

Use the **Define Fusion Accounting Hub Configuration for Rapid Implementation** task list to streamline your set-up configurations. The rapid implementation task list minimizes the time used to complete your key configurations and enables the day-to-day use of Oracle Fusion Accounting Hub applications.

The rapid implementation task list includes tasks that are:

- · Critical setup tasks.
- Setup steps that are required by most users.

To create an implementation project that includes the **Define Fusion Accounting Hub Configuration for Rapid Implementation** task list, use the **Manage Implementation Projects** page in the **Setup and Maintenance** work area. You can customize the task list as needed and assign and track each task.

Note: You are not restricted to only the set-up configuration in the rapid implementation task list. You can manually add the standard Fusion Accounting Hub offering task lists and tasks to your rapid implementation project.

Task Lists

The Define Fusion Accounting Hub Configuration for Rapid Implementation task list contains the following task lists:

Task List	Description
Define Enterprise Structures Configuration for Rapid Implementation	Define the enterprise structures configuration that includes financial and enterprise structures.
Define Accounting Entry Configuration for Rapid Implementation	Define the accounting entry configuration includes that accounting transformation rules for transactions sourced from external applications.
Define Ledger Configuration for Rapid Implementation	Define the configuration that includes general ledger and intercompany.
Define Financial Reporting Center Configuration for Rapid Implementation	Define the configuration that includes set up for financial reporting and integration with planning and financial management applications.
Define Financials Security Configuration for Rapid Implementation	Define the configuration for user and data roles set up.



Note: The Open First Period task is a required task and is part of the Define Fusion Accounting Hub Configuration for Rapid Implementation task list.

Creating an Oracle Fusion Accounting Hub Rapid Implementation Project: Worked Example

This example shows how to create an implementation project using the Oracle Fusion Accounting Hub rapid implementation task list.

The following table summarizes key decisions:

Decision to Consider	In This Example
What Oracle Fusion applications are included in this implementation?	The Oracle Fusion applications being implemented are: General Ledger Subledger Accounting Financial Reporting Center
Are the set-up requirements unique to this organization?	No
Can the rapid implementation task lists and tasks be used for this implementation?	Yes

Creating the Implementation Project

- 1. Navigator > Setup and Maintenance work area > Manage Implementation Projects.
- 2. Click Create.
- 3. Enter an unique name in the Name field.
- 4. Enter a **Description**.
- 5. Save and Open Project.
- 6. Add.
- 7. On the Select and Add: Task Lists and Task dialog window enter %Rapid Implementation% in the Name field. Use wildcard characters if you do not know the exact name of the task list or task.
- 8. Search
- 9. Select Define Fusion Accounting Hub Configuration for Rapid Implementation.
- 10. Done.
- 11. Expand the task list to see the task lists and tasks associated with your implementation project.
- 12. Done.



Oracle Fusion Accounting Hub Features: Overview

Oracle Fusion Accounting Hub (FAH) provides a complete set of accounting tools and unparalleled access to financial data, including:

- Oracle Fusion General Ledger features that provide:
 - Journal entry import and creation
 - Real time balances from a balances cube
 - Accounting controls
 - Close functionality
 - Cross-ledger intercompany balancing
 - o Calculation Manager for the definition of allocation rules using complex formulas
 - Automatic generation of allocation journals
 - Enhanced journal approval
 - Year-end process management
- Oracle Fusion Financial Reporting that provides:
 - Embedded balances cube functionality
 - o Multidimensional, online analytical processing (OLAP) of financial information
 - Ability to slice and dice data across dimensions
 - Drill up, down, and across on any parent level within the chart of accounts
- Oracle Fusion Subledger Accounting rules that provide flexible transformation of transaction and reference
 information from diverse, non-Oracle, industry applications into accurate, detailed, auditable accounting. Subledger
 Accounting is embedded in the Oracle Fusion subledgers such as Payables, Receivables, Assets, and Inventory.
- Oracle Hyperion Data Relationship Management (DRM) integration that provides
 - Ability to perform charts of accounts and hierarchies maintenance
 - Corporate wide accounting structures for Oracle and non-Oracle ledgers
 - Updates of charts of accounts and hierarchies across multiple ledgers
- Applications coexistence integration with the Oracle E-Business Suite (EBS) and Oracle PeopleSoft General Ledgers
 that provides the opportunity to leverage the Oracle Fusion Accounting Hub with minimal implementation effort.
 Continue to use your EBS or PeopleSoft applications for transaction processing, while taking advantage of the
 Oracle Fusion Accounting Hub for financial reporting and analysis.
- Oracle Hyperion Planning and Oracle Hyperion Financial Management integration that provides the opportunity to perform your planning, budgeting, and consolidation functions in an integrated accounting environment.
- Note: Oracle Hyperion Data Relationship Management, Planning, and Financial Management products all require licensing.



Oracle Fusion Accounting Hub: How It Works

The Oracle Fusion Accounting Hub process begins by using financial data from any or all of the following:

- Non-Oracle external applications including transaction and reference information from industry-specific applications.
- Oracle Fusion subledgers including subledger journals.
- Oracle E-Business Suite (EBS), Oracle PeopleSoft, and Oracle JD Edwards General Ledgers including account balances.

The Accounting Hub process ends with complete reporting and analysis solutions.

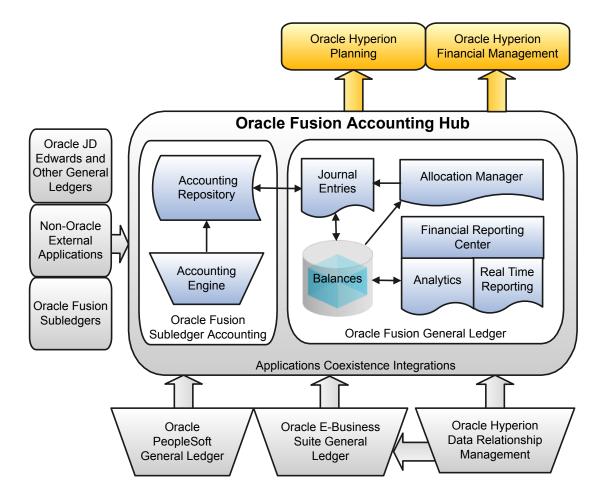
The Oracle Fusion Accounting Hub Components

The Accounting Hub contains the following components:

- Oracle Fusion Subledger Accounting to perform accounting transformations on external system data.
- Oracle Fusion General Ledger to enter and post journals including allocations.
- Oracle Fusion Financial Reporting Center for reporting and analysis.
- Integration with Oracle Hyperion Data Relationship Management (DRM) for chart of accounts and hierarchy maintenance.
- Applications coexistence integration with the EBS and PeopleSoft General Ledgers.



Integration with Oracle Hyperion Planning and Oracle Hyperion Financial Management.



Oracle Fusion Subledger Accounting for External Systems

The Oracle Fusion Accounting Hub enables Subledger Accounting to perform accounting transformations on external, non-Oracle system data. Subledger Accounting is also used to flexibly create accounting for Oracle subledgers such as Oracle Payables and Receivables. Subledger Accounting includes:

- Registration of your external systems, indicating what types of transactions or activities require accounting from those systems.
- Create a library of transaction and reference information that is used for defining accounting treatments.
- Configurable accounting rules to define accounting treatments for transactions.
- Accounting engine that combines transaction and reference information from source systems with accounting rules to create detailed journals stored in an accounting repository.
- Detailed subledger accounting journal entry repository to audit and reconcile accounting balances.
- Reports and user interface inquiries for analyzing accounting transformations.
- Configurable analytical balances based upon source system or reference attributes.



Oracle Fusion General Ledger

The Oracle Fusion General Ledger combines traditional general ledger functionality with embedded Oracle Hyperion Essbase functionality. The General Ledger functionality includes:

- Journal entry creation, including updating of account balances cubes and tables.
- Date effective trees for chart of accounts maintenance and financial reporting, including what if analysis.
- Automatic balances cube creation from the chart of accounts configuration, simplifying implementation.
- Preaggregated balances at every summarization level across each dimension of the chart of accounts and accounting periods to improve reporting performance.
- Multidimensional analysis using dimensions, such as the chart of accounts, periods, and currency, to provide drill down and drill through functionally.
- Drill to details and source ledger balances from summary balances.
- Intelligence and analytics embedded within the context of a journal entry, enabling quick and accurate completion of
 the journal entry process. As journal entries are entered, a what if analysis to determine the impact of the unposted
 journals on account balances is displayed in the user interface. The what if analysis eliminates navigating to an inquiry
 page or running a report to verify the results.
- Three balancing segments available in your chart of accounts for more detailed reporting.
- Automatic intercompany balancing journal creation in both Subledger Accounting and General Ledger applications, ensuring proper recording of transactions across legal entities.
- Calculation Manager rules using complex formulas to distribute revenue and costs throughout the organization for consistent periodic generation of allocation journal entries.

Financial Reporting Center

The Oracle Fusion General Ledger provides a Financial Reporting Center with robust financial reporting and analysis using data from your balances cubes. The dimensions contained in your chart of account segments become the direct source of multidimensional analysis. Direct links are maintained to your transactional data permitting comprehensive drill down from journals to transaction details. Use the following tools for your reporting and analysis:

- Financial Reporting to generate your reports.
- Smart View to generate spreadsheet reports.
- Oracle Fusion Transaction Business Intelligence to report using embedded analytics
- General Accounting and Journals dashboards to perform online inquiry and dashboard publication
- Account Monitor and Account Inspector to perform online multidimensional analysis of accounting balances

The Oracle Fusion Accounting Hub fits into Oracle's broader enterprise performance management framework through integration with Oracle Hyperion Planning and Oracle Hyperion Financial Management, You can perform:

- Financial consolidations using Financial Management.
- Planning and budgeting using Planning.

Integration with Hyperion Data Relationship Management

Oracle Fusion Accounting Hub is integrated with DRM, which is a master data management solution for creating and maintaining hierarchies across your enterprise. This integration maintains your charts of accounts values and hierarchies



in one central location. DRM synchronizes your hierarchies in Oracle Fusion and EBS General Ledgers. With licensing and integration of DRM, you can:

- Update and change DRM versions and hierarchies. The results of these changes are exported to the values sets in the Oracle Fusion and EBS General Ledgers, EBS parent child hierarchies, and Oracle Fusion Account Hierarchies (Trees).
- Synchronize charts of accounts and hierarchies across multiple Oracle general ledger instances.
- Store segment value attributes, such as account type, start date, and end date. Storing the values, provides the ability to enter these values in DRM and integrate them to Oracle Fusion and EBS General Ledgers.
- Submit standardized EBS standard requests and Oracle Fusion Enterprise processes from both the EBS and the Oracle Fusion General Ledgers to import values from DRM.

Integration with E-Business Suite and PeopleSoft General Ledgers

Integration between the EBS, Oracle PeopleSoft, and Oracle Fusion General Ledgers can be used to support an applications coexistence strategy. This strategy permits you to continue to use your EBS or PeopleSoft applications while also using the Oracle Fusion Accounting Hub. Application coexistence provides the ability to:

- Expand your reporting and analytical capabilities using the Financial Reporting Center and dashboard features of
 the Oracle Fusion Accounting Hub. This functionality includes the ability to drill down to EBS or PeopleSoft account
 balances and transactions reflected in the Oracle Fusion reports and dashboards.
- Continue to use your EBS or PeopleSoft applications for your procurement, payables, receivables, and any other EBS or PeopleSoft subledger processes without disruptions.

Accounting Configuration Offerings: Overview

The Setup and Maintenance work area in the Oracle Fusion Applications is used to manage the configuration of legal entities, ledgers, and reporting currencies. To create a legal entity or ledger, first create an implementation project. This implementation project can be populated by either adding a financials related offering or one or more task lists.

Note: Setup tasks that are not related to the ledger or legal entity setup tasks are opened from either an implementation project or directly from the Setup and Maintenance work area.

The financial applications have two predefined implementations:

- The Oracle Fusion Accounting Hub offering: Used to add the Oracle Fusion General Ledger and Oracle Fusion Subledger Accounting application features to an existing enterprise resource planning (ERP) system to enhance the reporting and analysis.
- The Oracle Fusion Financials offering includes the Oracle Fusion General Ledger and Oracle Fusion Subledger Accounting application features and one or more subledger financial applications.

When adding an offering to an implementation project, customize the tasks displayed by adding additional tasks.

Related Topics

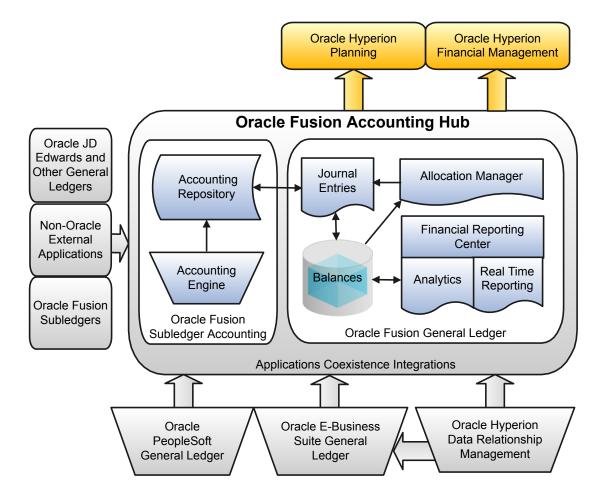
- What's an implementation project?
- What's a functional area?
- What's an offering?



Implementation Options for Oracle Fusion Accounting Hub: Overview

The Oracle Fusion Accounting Hub has three possible implementation scenarios that can be used separately or in conjunction with each other. These three scenarios are:

- Coexistence with Oracle E-Business Suite and other general ledgers providing enhanced allocations, reporting, and analytical functionality.
- Accounting for diverse business system events and transactions with configurable and auditable rule based accounting transformations.
- Expansion of financial reporting enabling creation of multidimensional, self service reporting and analytics with real-time accounting information from a single source with flexible formatting options.





Coexistence with E-Business Suite and PeopleSoft

Oracle Fusion Accounting Hub provides embedded integration with the Oracle E-Business Suite and Oracle PeopleSoft General Ledgers. The integration has three main components:

- Transfer of accounting balances from the E-Business Suite and PeopleSoft General Ledgers to the Oracle Fusion Accounting Hub General Ledger.
- Drill down from the Oracle Fusion Accounting Hub General Ledger to the E-Business Suite and PeopleSoft General Ledgers.
- Synchronization of charts of accounts segment values and hierarchies using Oracle Data Relationship Management, with E-Business Suite and Oracle Fusion General Ledger. Licensing of Data Relationship Management is required.

See the following guides for details on implementing this scenario:

- Oracle General Ledger Implementation Guide Release 12.2 for Oracle E-Business Suite General Ledger
- Oracle Data Relationship Management User's Guide
- Oracle Data Relationship Management User's Guide

Accounting for Diverse Transactions

The Oracle Fusion Accounting Hub provides accounting for diverse business system events and transactions using configurable and auditable rule based accounting transformations. The Accounting Hub process:

- Begins with transactions and data from your non-Oracle, industry specific subledgers and other general ledgers.
- Generates accounting events using the Create Accounting process to create journals that can be posted to the Oracle Fusion General Ledger from specific subledgers and other general ledgers.
- Ends with complete reporting and analysis solutions.

See the following for details on implementing this scenario: Define Accounting Transformation Configuration chapter in the Oracle Fusion Accounting Hub Implementation Guide.

Expansion of Financial Reporting

Oracle Fusion Accounting Hub enables creation of multidimensional, self service reporting and analytics with real time accounting information from a single source with flexible formatting options. The Financial Reporting Center includes:

- Predefined financial reports
- Ability to create new reports,
- Flexible options for dissemination of such reports
- Account Monitor and Account Inspector providing drill down and pivot views of your balances
- Oracle Smart View providing spreadsheet analytics
- Oracle Transaction Business Intelligence Oracle Business Intelligence Publisher provide key performance indicators (KPI), dashboards, and flexible reporting



See the following guides for details on implementing this scenario:

- Oracle Data Relationship Management Administrator's Guide
- Oracle Enterprise Performance Management Workspace Administrator's Guide
- Oracle Essbase Database Administrator's Guide
- Oracle Financial Reporting Administrator's Guide
- · Oracle Financial Reporting Studio User's Guide
- Oracle Fusion Accounting Hub Implementation Guide
- Oracle Smart View for Office User's Guide

Related Topics

- Oracle Financial Reporting Studio User's Guide
- Financial Reporting Center: How It Works
- Oracle Data Relationship Management Administrator's Guide

Setup Data Import and Export for Oracle Fusion Accounting Hub: Explained

The following sections describe the manual tasks that you must complete as part of the setup data export and import processes for the Oracle Fusion Accounting Hub offering.

Section	Description
Setup tasks performed before export	You must perform these tasks prior to initiating export processes for the Fusion Accounting Hub offering.
Setup tasks performed before import	You must perform these tasks prior to initiating import processes for the Fusion Accounting Hub offering.
Setup tasks performed after import	Setup data for these tasks is not imported from the source instance. Review and perform the manual setups steps on the target instance as required.

Refer to the Oracle Fusion Functional Setup Manager User's Guide for the steps to perform setup data export and import processes.

Setup Tasks Performed Before Export

Prior to initiating export processes from the source instance, you must verify and update setup for the following task. This is required to ensure that data is correctly imported into the target instance.

Activity Name	Task Name	Setup Steps
Define Chart of Accounts	Manage Account Combinations	Account combinations are not exported from the source instance. Before exporting, navigate to the Manage Chart of Accounts



Activity Name	Task Name	Setup Steps
		Instance page on the source instance and verify that dynamic insertion is enabled for your charts of accounts. As long as dynamic insertion is enabled, account combinations are created automatically as needed on the target instance.

Setup Tasks Performed Before Import

Prior to initiating import processes into the target instance, you must verify and update setup for the following task.

Activity Name	Task Name	Setup Steps
Define Common Applications Configuration for Fusion Accounting Hub	Define Implementation Users	The import process does not include implementation users and roles associated with them. For more information, see Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management (Oracle Fusion Applications Edition).

Setup Tasks Performed After Import

Setup data for the following tasks will not be imported. Review these tasks for relevance to your implementation. For relevant tasks, access the corresponding setup pages from your implementation project to create the setup on the target instance as needed.

Activity Name	Task Name
Define General Ledger Options	Manage Journal Approval RulesManage Document Sequences
Define Document Sequences	Manage Document Sequences
Define Approval Management for Financials	Manage Task Configurations for FinancialsManage Approval Groups for Financials

Setup data for the following tasks will not be imported from the source instance. Review the steps in the following table to create the setup on the target instance as needed.

Activity Name	Task Name	Setup Steps
Define Accounting Transformation Configuration	Import Supporting Reference Initial Balances	This task allows upload of initial subledger balances for supporting references. These balances are not imported from the source instance, but can be loaded directly to the target instance.



Activity Name	Task Name	Setup Steps
		For more information, see Oracle Fusion Accounting Hub Implementation Guide, Define Accounting Transformation.
Define Financial Reporting	Configure Smart View Client for Users	Manually reconfigure the Smart View client to point to the production instance.
		For more information about configuring the Smart View client for users, see:
		 Oracle Fusion Accounting Hub Implementation Guide, Define Financial Reporting
		 Oracle Enterprise Performance Management System Installation and Configuration Guide for Oracle Enterprise Performance Management, Installing Smart View and other topics
		 Oracle Hyperion Smart View for Office User's Guide for Oracle Hyperion Smart View
Define Financial Reporting Center Configuration	Define Essbase Database Connection in Workspace	Manually reconfigure the Essbase database connection in Hyperion Workspace.
		For more information about configuring the Hyperion Workspace Database Connection, see Oracle Fusion Accounting Hub Implementation Guide, Define Financial Reporting,
Define Financial Reporting Center Configuration	Configure Financial Reporting Studio Client for Users	Manually reconfigure the Financial Reporting Studio client to point to the production instance.
		For more information about configuring the Financial Reporting Studio client for users, see:
		Oracle Fusion Accounting Hub Implementation Guide, Define Financial Reporting.
		Oracle Enterprise Performance Management System Installation and Configuration Guide for Oracle Enterprise Performance Management. Refer to the following topics:
		 Installing Financial Reporting Studio and Financial Reporting Print Server
		Configuring the Financial Reporting Print Server
		 Administrative Information for Financial Reporting



Activity Name	Task Name	Setup Steps
Define Financial Reporting	Create Financial Statements	Export the financial report definitions from Workspace in the source environment.
		When exporting, you can export a single report, multiple reports in a zip file, or an entire folder structure in a zip file.
		 Navigator > General Accounting: Financial Reporting Center > select link: Open Workspace for Financial Reports. Navigate to Applications > BI Catalog. Navigate to File > Export.
		4. Save the file to the local desktop.
		Import the file into Workspace in the target environment.
		Navigator > General Accounting: Financial Reporting Center > select link: Open Workspace for Financial Reports.
		 Navigate to Application > BI Catalog. Navigate to File > Import. Select the file you had saved during the export.
		If you import the folder structure, the entire structure from the source instance is imported into the existing structure on the target instance. This could result in some redundant folders. In this case, you can reorganize child folders in the structure on the target instance and delete any unneeded folders.
Define Period Close Components	Manage Allocations and Periodic Entries	Export the allocation rules, rule sets, variables, and runtime prompt definitions from Calculation Manager in the source environment.
		When exporting, you can export at the application level or at a single rule or rule set level.
		Note: You must export and import rules for each application on the Essbase server separately.
		 Navigator > General Accounting: Journals > select link: Create Allocation Rules. Navigate to File > Export. Save the file to the local desktop.



Activity Name	Task Name	Setup Steps
		Import the file into Calculation Manager in the target environment.
		 Navigator > General Accounting: Journals > select link: Create Allocation Rules. Select the specific application in which to import the rules. Navigate to File > Import. Select the file you saved during the export.
Define Applications Coexistence Configuration for E-Business Suite	Manage Calendar Mappings for E-Business Suite	The import process does not include the calendar mappings of source Oracle E-Business Suite accounting calendars to target Oracle Fusion accounting calendars. If you use a calendar mapping, you must create it manually in your production Oracle E-Business Suite instance. For more information, see Oracle Fusion Accounting Hub Implementation Guide, Define Applications Coexistence Configuration.
Define Applications Coexistence Configuration for E-Business Suite	Manage Ledger Mappings for E-Business Suite	The import process does not include the mappings of source Oracle E-Business Suite ledgers to target Oracle Fusion ledgers. You must create the ledger mappings manually in your production Oracle E-Business Suite instance.
		For more information, see Oracle Fusion Accounting Hub Implementation Guide, Define Applications Coexistence Configuration.
Define Applications Coexistence Configuration for E-Business Suite	Register Applications Coexistence Instances for E-Business Suite	The import process does not include the registration of source Oracle E-Business Suite instances. You must manually register your source Oracle E-Business Suite instance or instances in the Oracle Fusion target instance.
		For more information, see Oracle Fusion Accounting Hub Implementation Guide. Define Applications Coexistence Configuration.
Define Hyperion Financial Management Integration	Define Hyperion Financial Management Configuration	Manually import the rules using the Oracle Enterprise Performance Management Life Cycle Management tool.
		For more information, see the Oracle Hyperion Artifact Life Cycle Management Utility User's Guide.



Activity Name	Task Name	Setup Steps
Define Hyperion Financial Management Integration	Define ERP Integrator Configuration for Hyperion Financial Management	Manually import the rules using the Oracle Enterprise Performance Management Life Cycle Management tool.
		For more information, see the Oracle Hyperion Artifact Life Cycle Management Utility User's Guide.
Define Budget Configuration	Define Budget Configuration in Hyperion Planning	Manually import the rules using the Oracle Enterprise Performance Management Life Cycle Management tool.
		For more information, see the Oracle Hyperion Artifact Life Cycle Management Utility User's Guide.
Define Budget Configuration	Define ERP Integrator Configuration for Hyperion Planning	Manually import the rules using the Oracle Enterprise Performance Management Life Cycle Management tool.
		For more information, see the Oracle Hyperion Artifact Life Cycle Management Utility User's Guide.

Manage Application Implementation

Setup and Maintenance: Overview

Oracle Functional Setup Manager enables rapid and efficient planning, configuration, implementation, deployment, and ongoing maintenance of Oracle Applications through self-service administration.

All Oracle Functional Setup Manager functionality is available from the Setup and Maintenance work area, which offers you the following benefits:

Self-Service Administration:

Manage all aspects of functional setup of Oracle Fusion applications at the business user level with an integrated, guided process for planning, configuration, implementation, deployment, and maintenance.

Configurable and Extensible:

Configure and Extend prepackaged list of tasks for setting up Oracle Fusion applications to better fit your business requirements.

Complete Transparency:

Get full visibility of Oracle Fusion applications end-to-end setup requirements with auto-generated, sequential task lists that include prerequisites and address dependencies.

Prepackaged Lists of Implementation Tasks:



Task lists can be easily configured and extended to better fit with business requirements. Autogenerated, sequential task lists include prerequisites and address dependencies to give full visibility to end-to-end setup requirements of Oracle Applications.

Rapid Start:

Specific implementations can become templates to facilitate reuse and rapid-start for comparable Oracle Applications across many instances.

Comprehensive Reporting:

A set of built-in reports helps to analyze, validate and audit configurations, implementations, and setup data of Oracle Applications.

With Oracle Functional Setup Manager you can:

- Learn about and analyze implementation requirements.
- Configure Oracle Applications to match your business needs.
- Achieve complete visibility to set up requirements through guided, sequential task lists downloadable into Excel for project planning.
- Enter setup data through easy-to-use user interfaces available directly from the task lists.
- Export and import data from one instance to another for rapid setup.
- Validate setup by reviewing setup data reports.
- Implement all Oracle Applications through a standard and consistent process.

Implementation Projects: Explained

You can create implementation projects to manage the implementation of an offering and functional areas as a unit throughout the implementation life cycle, or maintain the setup of specific business processes and activities customizing the list of tasks to complete their implementation.

An implementation project is the list of setup tasks you need to complete to implement selected offerings and functional areas. You create a project either by:

- Selecting an offering and its functional areas you want to implement together, then customize the list of tasks for such offering and functional areas as applicable.
- Selecting specific setup task lists and tasks you require for a specific configuration.

You can also assign these tasks to users and track their completion using the included project management tools.

Selecting Offerings

When creating an implementation project you see the list of offerings and functional areas that are configured for implementation. Implementation managers specify which of those offerings and functional areas to include in an implementation project. It is strongly recommended that you limit your selection to one offering per implementation project, even though the application does not prevent you from including more than one. The implementation managers should decide based on how they plan to manage their implementations. For example, if you implement and deploy different offerings at different times, then having separate implementation projects help to manage the implementation life cycles. Furthermore, the more offerings you included in an implementation project, the bigger the generated task list is. The implementation task list includes all setup tasks needed to implement all included offerings. Alternatively, segmenting into



multiple implementation projects makes the process easier to manage and ensures that import and export sequence of the project data is straightforward in the correct sequence.

Offerings: Explained

Offerings are application solution sets representing one or more business processes and activities that you typically provision and implement as a unit. They are, therefore, the primary drivers of functional setup of Oracle Fusion applications. Some of the examples of offerings are Financials, Procurement, Sales, Marketing, Order Orchestration, and Workforce Deployment. An offering is the highest level grouping of Oracle Fusion Applications functionality. They include functional areas, and alternative business rules known as features.

Enabling Offerings: Explained

When planning your implementation, you decide what business processes your organization or company performs or supports. These decisions determine the offerings and functional areas you want to implement. You then configure the offerings and functional areas that support the activities your organization or company performs. During the configuration process, you specifically enable offerings and functional areas for use before you implement them.

Enabling Offerings and Functional Areas

Use the Setup and Maintenance work area to help decide which offerings to enable for implementation. Once you decide to use an offering, you can select the Configure button to choose the configuration details and enable the offering, associated functional areas, and features. All the base functional areas of an offering are automatically enabled for implementation when you enable the parent offering. You choose which optional functional areas to enable. The functional areas appear in an expandable and collapsible hierarchy to facilitate progressive decision making for implementation.

Enabling Features

Features are optional or alternative business rules or methods used to fine-tune business processes and activities supported by an offering or a functional area. If features are available for the offering or functional areas, you can enable them to help meet your business requirements, if desired. In general, the features are set with a default configuration based on their typical usage in most implementations. You should always review the available features for the offering and functional areas and select them as appropriate. Dependent features appear visible when the feature choice they depend on is selected for implementation.

Enabling Offerings: Procedure

You enable offerings to customize the functionality that matches the services you plan on implementing.

Enabling Offerings

To enable offerings, follow these steps.

- Open the Setup and Maintenance work area (Navigator > Setup and Maintenance).
- 2. In the Setup and Maintenance Offerings page, select the offering you're using, then click **Configure**.
- 3. In the Configure page, select the **Enable** check box for the offering. Also select the **Enable** check box for each of the functional areas you want to use.
- **4.** Click the Features icon for the offering or functional area you have enabled, then enable any features you require. Select **Done** when complete.



5. Select **Done** to return to the Offerings page then repeat the same steps for each of the offerings you are using.

Features: Explained

Offerings include optional or alternative business rules or methods called feature choices, used to fine-tune business processes and activities supported by an offering or a functional area. You make feature selections according to your business requirements to get the best fit with the offering. If the selected offerings and functional areas have dependent features then those features are applicable when you implement the corresponding offering or functional area.

Feature choices can be one of three different types:

Yes or No

If a feature can either be applicable or not be applicable to an implementation, a single check box is presented for selection. Check or deselect to specify yes or no respectively.

Single Select

If a feature has multiple choices but only one can be applicable to an implementation, multiple choices are presented as radio buttons. You can turn on only one of those choices.

Multi-Select

If the feature has multiple choices but one or more can be applicable to an implementation then all choices are presented with a check box. Select all that apply by checking the appropriate choices.



2 Define Enterprise Structures for Fusion Accounting Hub

Enterprise Structures: Overview

Oracle Fusion Applications have been designed to ensure your enterprise can be modeled to meet legal and management objectives. The decisions about your implementation of Oracle Fusion Applications are affected by your:

- Industry
- Business unit requirements for autonomy
- Business and accounting policies
- Business functions performed by business units and optionally, centralized in shared service centers
- · Locations of facilities

Every enterprise has three fundamental structures, that describe its operations and provide a basis for reporting.

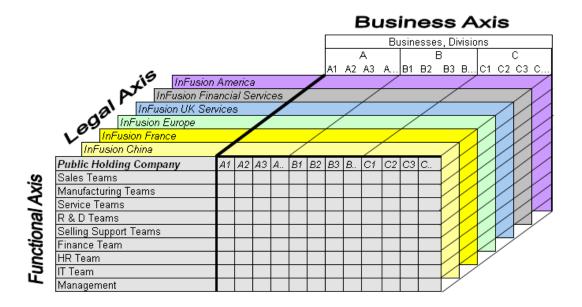
- Legal
- Managerial
- Functional

In Oracle Fusion, these structures are implemented using the chart of accounts and organization hierarchies. Many alternative hierarchies can be implemented and used for reporting. You are likely to have one primary structure that organizes your business into:

- Divisions
- Business Units
- Departments



Aligned these structures with your strategic objectives.



Legal Structure

The figure above shows a typical group of legal entities, operating various business and functional organizations. Your ability to buy and sell, own, and employ comes from your charter in the legal system. A corporation is:

- A distinct legal entity from its owners and managers.
- Owned by its shareholders, who may be individuals or other corporations.

Many other kinds of legal entities exist, such as sole proprietorships, partnerships, and government agencies.

A legally recognized entity can own and trade assets and employ people in the jurisdiction in which the entity is registered. When granted these privileges, legal entities are also assigned responsibilities to:

- Account for themselves to the public through statutory and external reporting.
- Comply with legislation and regulations.
- Pay income and transaction taxes.
- Process value added tax (VAT) collection on behalf of the taxing authority.

Many large enterprises isolate risk and optimize taxes by incorporating subsidiaries. They create legal entities to facilitate legal compliance, segregate operations, optimize taxes, complete contractual relationships, and isolate risk. Enterprises use legal entities to establish their enterprise's identity under the laws of each country in which their enterprise operates.

In the figure above:

- A separate card represents a series of registered companies.
- Each company, including the public holding company, InFusion America, must be registered in the countries where they do business.



• Each company contributes to various divisions created for purposes of management reporting. These are shown as vertical columns on each card.

For example, a group might have a separate company for each business in the United States (US), but have its United Kingdom (UK) legal entity represent all businesses in that country.

The divisions are linked across the cards so that a business can appear on some or all of the cards. For example, the air quality monitoring systems business might be operated by the US, UK, and France companies. The list of business divisions is on the Business Axis.

Each company's card is also horizontally striped by functional groups, such as the sales team and the finance team. This functional list is called the Functional Axis. The overall image suggests that information might, at a minimum, be tracked by company, business, division, and function in a group environment. In Oracle Fusion Applications, the legal structure is implemented using legal entities.

Management Structure

Successfully managing multiple businesses requires that you segregate them by their strategic objectives, and measure their results. Although related to your legal structure, the business organizational hierarchies do not have to be reflected directly in the legal structure of the enterprise. The management structure can include divisions, subdivisions, lines of business, strategic business units, profit, and cost centers. In the figure above, the management structure is shown on the Business Axis. In Oracle Fusion Applications, the management structure is implemented using divisions and business units as well as being reflected in the chart of accounts.

Functional Structure

Straddling the legal and business organizations is a functional organization structured around people and their competencies. For example, sales, manufacturing, and service teams are functional organizations. This functional structure is represented by the Functional Axis in the figure above. You reflect the efforts and expenses of your functional organizations directly on the income statement. Organizations must manage and report revenues, cost of sales, and functional expenses such as research and development and selling, general, and administrative expenses. In Oracle Fusion Applications, the functional structure is implemented using departments and organizations, including sales, marketing, project, cost, and inventory organizations.

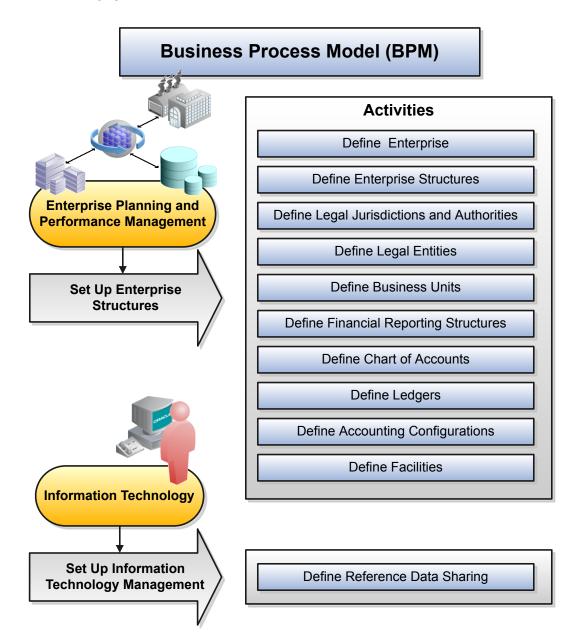
Enterprise Structures Business Process Model: Explained

In Oracle Fusion Applications, the Enterprise Performance and Planning Business Process Model illustrates the major implementation tasks that you perform to create your enterprise structures. This process includes:

- Set Up Enterprise Structures business process, which consists of implementation activities that span many product families.
- Information Technology, a second Business Process Model which contains the Set Up Information Technology Management business process.
- Define Reference Data Sharing, which is one of the activities in this business process and is important in the
 implementation of the enterprise structures. This activity creates the mechanism to share reference data sets across
 multiple ledgers, business units, and warehouses, reducing the administrative burden and decreasing the time to
 implement.



The following figure and chart describes the Business Process Model structures and activities.



BPM Activities	Description
Define Enterprise	Define the enterprise to get the name of the deploying enterprise and the location of the headquarters.
Define Enterprise Structures	Define enterprise structures to represent an organization with one or more legal entities under common control. Define organizations to represent each area of business within the enterprise.



BPM Activities	Description
Define Legal Jurisdictions and Authorities	Define information for governing bodies that operate within a jurisdiction.
Define Legal Entities	Define legal entities and legal reporting units for business activities handled by the Oracle Fusion Applications.
Define Business Units	Define business units of an enterprise to perform one or many business functions that can be rolled up in a management hierarchy. A business unit can process transactions on behalf of many legal entities. Normally, it has a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss.
Define Financial Reporting Structures	Define financial reporting structures, including organization structures, charts of accounts, organizational hierarchies, calendars, currencies and rates, ledgers, and document sequences which are used in organizing the financial data of a company.
Define Chart of Accounts	Define chart of accounts including hierarchies and values to enable tracking of financial transactions and reporting at legal entity, cost center, account, and other segment levels.
Define Ledgers	Define the primary accounting ledger and any secondary ledgers that provide an alternative accounting representation of the financial data.
Define Accounting Configurations	Define the accounting configuration that serves as a framework for how financial records are maintained for an organization.
Define Facilities	Define your manufacturing and storage facilities as Inventory Organizations if Oracle Fusion tracks inventory balances there and Item Organizations if Oracle Fusion only tracks the items used in the facility but not the balances.
Define Reference Data Sharing	Define how reference data in the applications is partitioned and shared.

Note: Some product-specific implementation activities are not listed here and depend on the applications you are implementing. For example, you can implement Define Enterprise Structures for Human Capital Management, Project Management, and Sales Management.

Global Enterprise Configuration: Points to Consider

Start your global enterprise structure configuration by discussing what your organization's reporting needs are and how to represent those needs in the Oracle Fusion Applications. The following are some questions and points to consider as you design your global enterprise structure in Oracle Fusion.

- Enterprise Configuration
- Business Unit Management
- Security Structure
- Compliance Requirements



Enterprise Configuration

- What is the level of configuration needed to achieve the reporting and accounting requirements?
- What components of your enterprise do you need to report on separately?
- Which components can be represented by building a hierarchy of values to provide reporting at both detail and summary levels?
- Where are you on the spectrum of centralization versus decentralization?

Business Unit Management

- What reporting do I need by business unit?
- How can you set up your departments or business unit accounts to achieve departmental hierarchies that report accurately on your lines of business?
- What reporting do you need to support the managers of your business units, and the executives who measure them?
- How often are business unit results aggregated?
- What level of reporting detail is required across business units?

Security Structure

- · What level of security and access is allowed?
- Are business unit managers and the people that report to them secured to transactions within their own business unit?
- Are the transactions for their business unit largely performed by a corporate department or shared service center?

Compliance Requirements

- How do you comply with your corporate external reporting requirements and local statutory reporting requirements?
- Do you tend to prefer a corporate first or an autonomous local approach?
- Where are you on a spectrum of centralization, very centralized or decentralized?

Modeling Your Enterprise Management Structure in Oracle Fusion: Example

This example uses a fictitious global company to demonstrate the analysis that can occur during the enterprise structure configuration planning process.



Scenario

Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion Enterprise Resource Planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers. You are chairing a committee to discuss creation of a model for your global enterprise structure including both your US and UK operations.

InFusion Corporation

InFusion Corporation has 400 plus employees and revenue of 120 million US dollars. Your product line includes all the components to build and maintain air quality monitoring (AQM) applications for homes and businesses. You have two distribution centers and three warehouses that share a common item master in the US and UK. Your financial services organization provides funding to your customers for the initial costs of these applications.

Analysis

The following are elements you must consider in creating your model for your global enterprise structure.

- Your company is required to report using US Generally Accepted Accounting Principles (GAAP) standards and UK Statements of Standard Accounting Practice and Financial Reporting Standards. How many ledgers do you want to achieve proper statutory reporting?
- Your managers need reports that show profit and loss (revenue and expenses) for their lines of business. Do you use business units and balancing segments to represent your divisions and businesses? Do you secure data by two segments in your chart of accounts which represents each department and legal entity? Or do you use one segment that represents both to produce useful, but confidential management reports?
- Your corporate management requires reports showing total organizational performance with drill-down capability to the supporting details. Do you need multiple balancing segment hierarchies to achieve proper rollup of balances for reporting requirements?
- Your company has all administrative, account payables, procurement, and Human Resources functions performed at their corporate headquarters. Do you need one or more business units in which to perform all these functions? How is your shared service center configured?

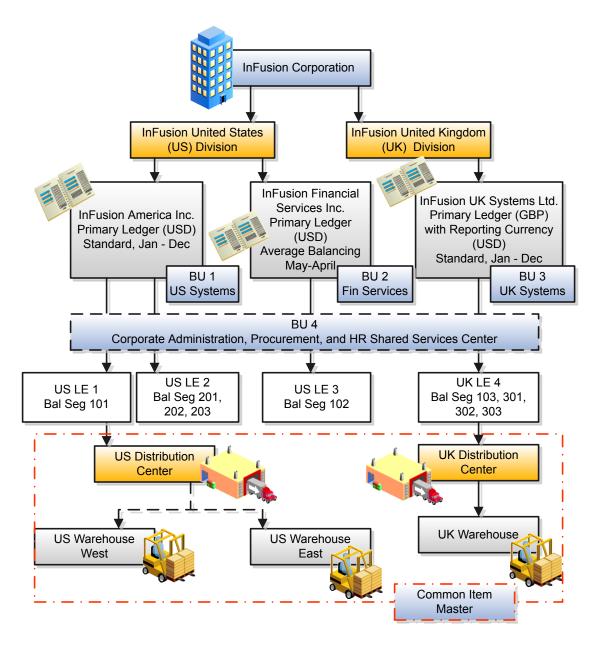
Global Enterprise Structure Model

The following figure and table summarize the model that your committee has designed and uses numeric values to provide a sample representation of your structure. The model includes the following recommendations:

- Creation of three separate ledgers representing your separate legal entities:
 - InFusion America Inc.
 - InFusion Financial Services Inc.
 - InFusion UK Services Ltd.



- Consolidation of results for application components, installations, and maintenance product lines across the enterprise
- All UK general and administrative costs processed at the UK headquarters
- US Systems' general and administrative costs processed at US Corporate headquarters
- US Financial Services maintains its own payables and receivables departments





Real World Entity	Entity Name	Enterprise	Legal Entity	BSV	Ledger	BU	Cost Center	Dept	In ventory Org
Enterprise	InFusion Group								
Company	USLE1								
Company	USLE2		-	-					
Company	USLE3		- 2	7					
Company	UK LE 4		<u> </u>						
Business Unit	U S Systems BU 1								
Business Unit	FIN Services BU 2								
Bu siness Unit	UK Systems BU 3								
Division	In Fusion UK						•		
Division	InFusion U S						•		
Headquarters	BU 4								
Shared Service Center	BU 4								
Department	AP Department								
List of Items	Common Item Master								
Distribution Center	US Distribution Center								
Distribution Center	UK Distribution Center								
Warehouse	U S Warehouse West								
Warehouse	US Warehouse East								
Warehouse	UK Warehouse								
B SV = Balancing Segme	ent Value								
BU = Business Unit									
Dept = Department									
Org = Organization									
= Mandatory Setup									
Setup									

In this chart, the green globe stands for required and gold globe stands for optional setup. The following statements expand on the data in the chart.

- The enterprise is required because it serves as an umbrella for the entire implementation. All organizations are created within an enterprise.
- Legal entities are also required. They can be optionally mapped to balancing segment values or represented by ledgers. Mapping balancing segment values to legal entities is required if you plan to use the intercompany functionality. The InFusion Corporation is a legal entity but is not discussed in this example.
- At least one ledger is required in an implementation in which you record your accounting transactions.
- Business units are also required because financial transactions are processed in business units.
- A shared service center is optional, but if used, must be a business unit.
- Divisions are optional and can be represented with a hierarchy of cost centers or by a second balancing segment value.
- Departments are required because they track your employees.
- Optionally, add an item master organization and inventory organizations if you are tracking your inventory transactions in Oracle Fusion Applications.



Note: Some Oracle Fusion Human Capital Management and Oracle Sales Cloud implementations do not require recording accounting transactions and therefore, do not require a ledger.

Running Diagnostic Tests for Enterprise Structures Setup Data: Explained

You can run diagnostic tests to perform a health check and data validation on the following enterprise structures setup data:

- · Chart of Accounts
- Value Sets and Values
- Account Hierarchies Versions
- Accounting Calendars
- Legal Entities and Legal Reporting Units
- Ledgers Setup
- To access the Diagnostic Dashboard and execute the diagnostic tests, you must be granted the Application Diagnostics Regular User job role.
- Once you have been granted the role, then you can view the link to run the diagnostics tests under the
 Troubleshooting on Settings and Actions menu on your Welcome page.
- If you have the Application Diagnostics Viewer job role, you can view the diagnostic test results, but not run tests.

Use the following steps:

- To access the **Diagnostic Dashboard** and execute the diagnostic tests, you must be granted the **Application Diagnostics Regular User** job role.
- To view the link to run the diagnostics tests under the Troubleshooting on Settings and Actions menu on your Welcome page after you have been granted the user role.
- Note: If you have the **Application Diagnostics Viewer** job role, you can view the diagnostic test results, but not run tests.

Define Common Financials Configuration for Rapid Implementation

Enterprise Structures Rapid Implementation: Overview

You can use Rapid Implementation spreadsheet template to rapidly implement Oracle Fusion enterprise structures and set up the following objects:

Chart of accounts, including value sets, value set values, structures, and structure instances



- Account hierarchies including trees and tree versions
- Accounting calendars (monthly type only)
- · Legal entities including legal addresses, jurisdictions, and identifiers
- Primary ledgers with legal entity assignments to primary balancing segment values
- Business units
- Document and journal sequencing.

Below are the Functional Setup Manager tasks which are required to be performed to rapidly implement the Oracle Fusion General Ledger:

- Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheet: Downloads the rapid implementation excel spreadsheet template.
- **Upload Chart of Accounts**: Starts the **Create Enterprise Structure** process from the **Scheduled Process** page using the upload data file as a parameter. You must upload your first file, ChartOfAccounts.xml, is created using the template. Monitor the process for the successful completion.
- **Deploy Chart of Accounts**: Opens the **Manage Chart of Accounts Instance** page. Select the accounting flexfield and press the **Deploy** button to deploy the accounting flexfield.
- Upload Ledger, Legal Entities, and Business Units: Starts the Create Enterprise Structure process from
 the Scheduled Process page with the upload data file as a parameter. You must upload your second file,
 FinancialsCommonEntities.xml, created using the template and monitor for the successful completion of the process.

The following are best practices and recommendations:

- Design your chart of accounts carefully, including the number and sequence of segments.
- Identify the aspects of your business that you want to track and analyze, such as company, division, cost center, department, account, and line of business.
- Anticipate future growth and reporting requirements by defining one or more future segments. Assign those segments with a default value to streamline data entry.
- Create account hierarchies to reflect managerial, legal, or geographical relationships between your value set values.
 Your financial balances are preaggregated at each parent level in your hierarchy, thus allowing fast and robust account inquiry and drill down.
- Note: Once you begin using your chart of accounts, making changes to the segments is neither recommended nor supported.

Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheets: Explained

Represent enterprise structures of your chart of accounts, ledger, legal entities, and business unit configuration to track and report on your financial objectives and meet your reporting requirements. These components are the underlying structure for organizing financial information and reporting.

The chart of accounts within the ledger facilitates:

- · Aggregating data from different operations, from within an operation, and from different business flows
- Consistent definitions to your stakeholders in compliance with legislative and corporate reporting standards and aids in management decisions



Rapid implementation is a way to configure a financial enterprise and financial reporting structures quickly using sheets in a workbook that upload lists of:

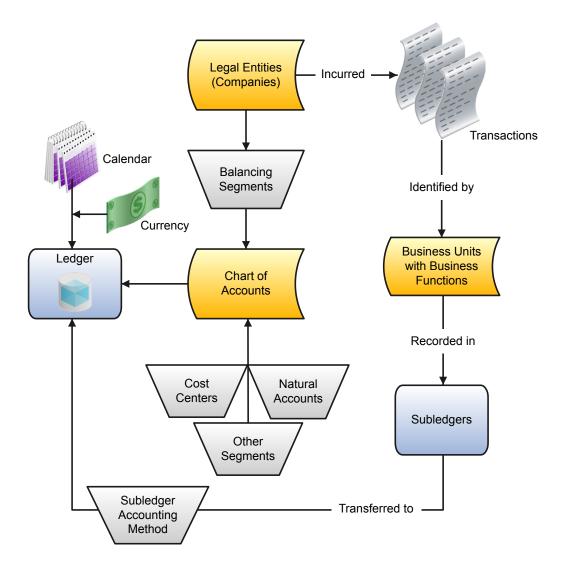
- Companies (legal entities)
- Ledgers by country
- Business units
- Chart of accounts and segment values
- · Segment value hierarchies
- Financial sequences
- Required subledger accounts

Once the sheets have been uploaded, the application creates your:

- Chart of accounts structure and instance
- Segment value hierarchies
- Key accounts such as retained earnings
- Required subledger accounts
- Calendar
- Primary ledgers by country
- Legal entities and their locations
- Business units
- Document and journal sequences



The following figure shows the relationship of these components.



- **Legal Entity**: Identifies a recognized party with rights and responsibilities given by legislation, which has the right to own property and the responsibility to account for themselves.
- **Chart of Accounts**: Configures accounts consisting of components called segments that are used to record balances and organize your financial information and reporting.
- **Segment**: Contains a value set that provides formatting and validation of the set of values used with that segment. When combined, several segments create an account combination for recording your transactions and journal entries.
- **Segment Label**: Identifies certain segments in your chart of accounts and assigns special functionality to those segments. The required segment labels are:
 - Balancing Segment: Ensures that all journals balance for each balancing segment value or combination of multiple balancing segment values to use in financial processes and reporting. The three balancing segment labels are: Primary balancing segment, Second balancing segment, and Third balancing segment. The Primary balancing segment label is required and must be the first segment in the Rapid Implementation spreadsheet.



Natural Account: Facilitates processes in the General Ledger application, such as retained earnings posting.
 For each child value, you must assign an Account Type. You can select from one of the general choices to mark the account value as an Asset, Liability, Owner's Equity, Revenue, or Expense account.

If the account is used by the rapid implementation solution to provide accounts for setup objects, select the appropriate Expanded Account Type value for the child account. Examples of expanded account types required for setup objects are:

- Owner's Equity Retained Earnings: To set up General Ledger ledgers.
- Liability Accounts Payable: To set up Payables common options.
- Asset Accounts Receivable: To set up Receivables receipt methods.

Accounts tagged with expanded account types are automatically assigned a financial category. You can override the default category in the Financial Category field, or leave it out.

- **Cost Center**: Facilitates grouping of natural accounts by functional cost types, accommodating tracking of specific business expenses across natural accounts.
- **Ledger**: Maintains the records and is a required component in your configuration. The rapid implementation process:
 - Creates your primary ledgers by combining your chart of accounts, calendar, and currency as well as other required options defined in the sheets.
 - Assigns a default value for the fourth component, which is the subledger accounting method. The subledger accounting method is used to group subledger journal entry rule sets together to define a consistent accounting treatment.
 - Note: The standard accrual method is the default subledger accounting method assigned to the primary ledger.
 - Creates a General Ledger balances cube for each ledger with a unique chart of accounts and calendar combination. Each segment is created as a dimension in the balances cube along with the standard cube dimensions.
- Business Units with Business Functions: Identifies where subledger transactions are posted and provides
 access to perform subledger business processes. When configured, business units are assigned to a primary ledger
 and a default legal entity.
- **Subledger**: Captures detailed transactional information, such as supplier invoices, customer payments, and asset acquisitions. Uses subledger accounting to transfer transactional balances to the ledger where they are posted.
- Note: Segment Value Hierarchies: You can create more than one hierarchy for any of your chart of accounts segments during initial setup. You can also create additional hierarchies after the initial setup is done by uploading the rapid implementation spreadsheet data. Document and Journal Sequences: You can create sequences for each legal entity or ledger based on the predefined country defaults. Document sequences are created for:
 - Payables invoices
 - Payments
 - Receivables invoices
 - Receivables credit memos
 - Receivables adjustment activities

Reporting and accounting journal sequences are created for subledger journals and General Ledger journals.



Related Topics

Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheets: How They're Processed

Preparing and Loading Enterprise Structures Rapid Implementation Templates: Worked Example

Your company is implementing Oracle Fusion General Ledger using the rapid implementation methodology to create your enterprise structures.

Use the Microsoft Excel template to rapidly implement the following enterprise structures: chart of accounts, hierarchies, legal entities, accounting calendar, primary ledgers, business units, and journal and document sequences.

Gathering the Data

Start your implementation carefully, by first gathering the necessary data:

- 1. Determine your chart of accounts segments.
- 2. Compile your segment values and hierarchies data.
- 3. Identify your legal entities and business units.
- 4. Determine your retained earnings account.

Preparing the Template Data Sheets

Complete the data sheets for each enterprise structure. Character and word limitations exist and apply to Essbase and all Oracle Fusion General Ledger configurations. See topic: Essbase Character and Word Limitations.

1. Complete the COA, Calendar, and Ledger data sheet with your setup data for your chart of accounts, calendar, and ledger.

Entry Field	Comments
Name	Enter the name of the primary ledger. A primary ledger is created for each unique country entered on the Companies and Legal Entities sheet. For example: o One of the legal entities is based in United States and another in Canada. The name provided is Vision Ledger. Two primary ledgers, Vision Ledger US and Vision Ledger CA, are created.
Currency	Select the primary ledger currency from the list of values.
Period Frequency	Select from the following values:
	o Monthly
	o 4/4/5
	o 4/5/4
	o 4 week



Entry Field	Comments		
Adjusting Periods	Select from the following values.		
	_o None		
	Once at year end		
	Once at beginning of year and once at end of the year		
	Once at mid of year and once at year end		
	_o Quarterly		
Fiscal Year Start Date	Enter the start date of the accounting calendar using the format mm/dd/yyyy.		
Segment	Enter the segment name for each segment.		
Segment Label	Assign one of the following segment labels from the list of values to your segments.		
	o Primary, Second, and Third Balancing		
	 Second Balancing and Cost Center 		
	o Cost Center		
	o Natural Account		
	o Intercompany		
	o Local Use		
	Other considerations:		
	o Primary Balancing and Natural Account are required.		
	o Assign segment labels once.		
	The Intercompany segment is optional. If specified, the segment reuses the value set		
	created for the Primary Balancing segment. The Second and Third Balancing are optional. Second balancing segment may be assigned with the cost center segment.		
Short Prompt	Enter a short prompt for each segment.		
Display Width	Enter a display width for each segment.		
Add Segment Sheets button	After entering all your segments, click this button. A data sheet is created for each additional segment where you can enter the segment values and hierarchies. The segment value and hierarchies data entry sheets for your primary balancing segment and natural account segment are provided.		

2. Complete the Business Units data entry sheet if needed.



Entry Field	Comments	
Name	Enter the name of your business units (BUs). The process enables all the functions and sets the following:	
	。 Reference Data Set is Common .	
	$_{\mbox{\scriptsize o}}$ The primary ledger created in this process is the default ledger for all BUs.	
	 First entered legal entity on the Companies and Legal Entities data sheet is the default legal entity for all BUs. 	
Default Legal Entry Name	Determines the primary ledger assigned to the business unit.	

3. Complete the Companies and Legal Entities data entry sheet for your companies, hierarchies, and legal entity data.

Entry Field	Comments	
Parent9 to Parent1	To define hierarchies, enter the parent values in Parent9 to Parent1 column. The first-level parent is entered in the Parent1 column. The parent of Parent1 is entered in the column Parent2 and so on.	
Child	Enter all the child values or level0 values which are your postable account values in this column.	
	O Duplicate values are not allowed.	
	o Enter a required value and description on each row until the last row.	
	o Plan the hierarchies before you start entering them in this template.	
Company Description	Name of the company. For example, Vision Corporation, Vision US, and Vision UK.	
Legal Entity: Name	Optionally enter a legal entity name.	
Legal Entity: Identifier	Legal identifier such as a tax ID or government registration number.	
Legal Entity: Country	Select from the list of values provided. The identifying jurisdiction is derived from the country you select.	
Legal Entity: Address Line	Enter the address in comma-separated format.	
Legal Entity: City, State, County, Province, and Postal Code	Address for the legal entity.	
Legal Entity: Registration Number	Identifies legal entities registered for your company and recognized in law for which you want to record and perform transactions.	
Legal Entity: Reporting Unit Registration Number	Identifier for the lowest level component of a legal structure that requires registrations. Map the legal entity details to the primary balancing segment value. If the legal entity maps to multiple primary balancing segment values consecutively, only specify the legal entity for the first primary balancing segment value.	



4. Complete the Natural Accounts data entry sheet.

Entry Field	Comments
Parent9 to Parent1 and Child	Enter your parent values in these columns. First-level parents are entered in the column Parent1 and so on. Duplicate values are not allowed.
Description	Provide descriptions for each of the parent values as well as the child values.
Account Type	Categorizes the accounts. Account type is used to determine which account to use in application processes such as retained earnings calculation at the end of the year. The account type is also used in the upload processes to create general ledger and subledger (Payables and Receivables) accounts in some setup pages. Select the appropriate values from the list for each account.
Financial Category	An optional field that is used with Oracle Business Intelligence applications. The predefined values consist of the standard financial categories. For example: Accounts payable, Accounts receivable, Accrued liabilities, Accumulated depreciation, Cash, and Common stock.

- 5. To create more than one hierarchy for any of your chart of accounts segments during the initial setup.
 - **a.** Click the **Generate Additional Hierarchy** button on the segment's worksheet. This creates a new worksheet and populates it with the data already entered for that segment.
 - **b.** Change this data as required for the new hierarchy.
 - Note: For the company segment, adding legal entity information is not supported on this sheet.
 - Note: With this button you can create hierarchies, but not new versions of existing hierarchies.
- 6. Complete the Financial Sequences data entry sheet.

Entry Field	Comments		
Enabled	Enter Yes to enable the sequence.		
Restart	Enter the period of time used to restart the sequence. Select: o Annually o Monthly		
	o Never		
Initial Value	Enter a whole number greater than 1. All versions of the sequence start with the same number.		

7. Click the **Step 1**: **Validate** button to validate the workbook data.

Loading the Templates

- 1. After you finish preparing the data in the sheets, click the **Step 2**: **Generate Chart of Accounts File** button on the COA, Calendar, and Ledger sheet. The process generates an XML data file called **ChartOfAccounts.xml**.
- 2. Save the file in a local directory.
- 3. Click the **Step 3**: **Generate Ledger**, **LE**, **and BU File** button on the COA, Calendar, and Ledger sheet. The process generates an XML data file called **FinancialsCommonEntities.xml**.



- 4. Save the file in a local directory.
- 5. On the Setup and Maintenance page in the Functional Setup Manager, search for and select the **Upload Chart of Accounts** task. This task runs the Upload Enterprise Structures process.
- 6. Click the Upload File button and select the first file you saved, which is ChartOfAccounts.xml.
- 7. Click **Submit**. The Scheduled Processes page opens.
- 8. Verify that the process completes without errors or warnings.
- 9. Click Done.
- Select the **Deploy Chart of Accounts** task on the Setup and Maintenance page. Select the **Deploy** button for the
 accounting flexfield.
- 11. Refresh the page until the green check mark appears and verifies that the deployment was successful.
- **12.** On the Setup and Maintenance page, select the **Upload Ledger**, **Legal Entities**, **and Business Units** task. This runs the Upload Enterprise Structures process.
- 13. Click the Upload File button and select the second XML file you saved, which is FinancialsCommonEntities.xml.
- 14. Click **Submit**. The Scheduled Processes page opens.
- **15.** Verify that the process completed without errors or warnings.
- 16. Click Done.

Running Automatic Processes

- 1. The following processes are run automatically during the rapid implementation flow to configure the ledger and associated setup and create the balances cubes:
 - Submit Account Configuration, which includes general ledger balances cube creation
 - Online Audit of Hierarchy Data of tree and tree versions
 - Row and Column Flattening of hierarchy data
 - Publish Hierarchies

Related Topics

Essbase Character and Word Limitations

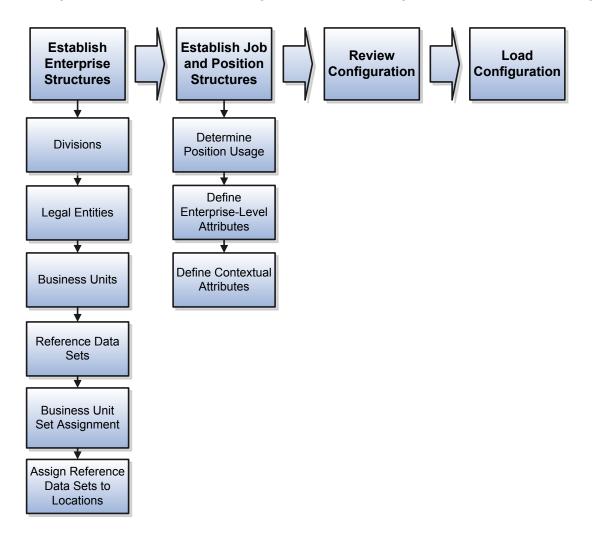
Define Initial Configuration with the Enterprise Structures Configurator

Establishing Enterprise Structures Using the Enterprise Structures Configurator: Explained

The Enterprise Structures Configurator is an interview-based tool that guides you through the process of setting up a basic enterprise structure. By answering questions about your enterprise, the tool creates a structure of divisions, legal entities, business units, and reference data sets that reflects your enterprise structure. After you create your enterprise structure, you also follow a guided process to determine whether to use positions, and whether to set up additional attributes for jobs and positions. After you define your enterprise structure and your job and position structures, you can review them, make any necessary changes, and then load the final configuration.



This figure illustrates the process to configure your enterprise using the Enterprise Structures Configurator.



To be able to use the Enterprise Structures Configurator, you must select the Enterprise Structures Guided Flow feature for your offerings on the Configure Offerings page in the Setup and Maintenance work area. If you don't select this feature, then you must set up your enterprise structure using individual tasks provided elsewhere in the offerings, and you can't create multiple configurations to compare different scenarios.

Establish Enterprise Structures

To define your enterprise structures, use the guided flow within the Establish Enterprise Structures task to enter basic information about your enterprise, such as the primary industry. You then create divisions, legal entities, business units, and reference data sets. The Establish Enterprise Structures task enables you to create multiple enterprise configurations so that you can compare different scenarios. Until you load a configuration, you can continue to create and edit multiple configurations until you arrive at one that best suits your enterprise.

Establish Job and Position Structures

You also use a guided process to determine whether you want to use jobs only, or jobs and positions. The primary industry that you select in the Establish Enterprise Structures task provides the application with enough information to make an initial recommendation. You can either accept the recommendation, or you can answer additional questions about how you



manage people in your enterprise, and then make a selection. After you select whether to use jobs or positions, you are prompted to set up a descriptive flexfield structure for jobs, and for positions if applicable. Descriptive flexfields enable you to get more information when you create jobs and positions.

Review Configuration

You can view a result of the interview process prior to loading the configuration. The review results, show the divisions, legal entities, business units, reference data sets, and the management reporting structure that the application will create when you load the configuration.

Load Configuration

You can load only one configuration. When you load a configuration, the application creates the divisions, legal entities, business units, and so on. After you load the configuration, you then use individual tasks to edit, add, and delete enterprise structures.

Designing an Enterprise Configuration: Example

This example illustrates how to set up an enterprise based on a global company operating mainly in the US and the UK with a single primary industry.

Scenario

InFusion Corporation is a multinational enterprise in the high technology industry with product lines that include all the components that are required to build and maintain air quality monitoring systems for homes and businesses. Its primary locations are in the US and the UK, but it has smaller outlets in France, Saudi Arabia, and the United Arab Emirates (UAE).

Enterprise Details

In the US, InFusion employs 400 people and has company revenue of 120 million US dollars.. Outside the US, InFusion employs 200 people and has revenue of 60 million US dollars.

Analysis

InFusion requires three divisions.

- The US division covers the US locations.
- The Europe division covers UK and France.
- Saudi Arabia and the UAE are covered by the Middle East division.

InFusion requires legal entities with legal employers, payroll statutory units, tax reporting units, and legislative data groups for the US, UK, France, Saudi Arabia, and UAE, to employ and pay its workers in those countries.

InFusion requires a number of departments across the enterprise for each area of business, such as sales and marketing, and a number of cost centers to track and report on the costs of those departments.

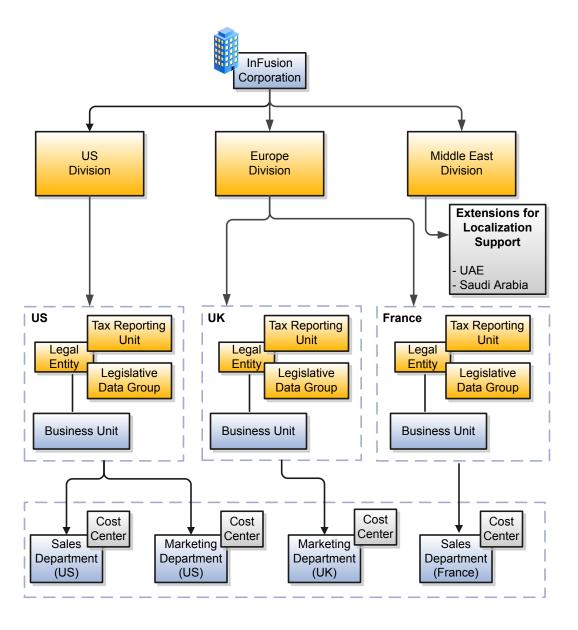
InFusion has general managers responsible for business units within each country. Those business units may share reference data. Some reference data can be defined within a reference data set that multiple business units may subscribe to. Business units are also required for financial purposes. Financial transactions are always processed within a business unit.



Resulting Enterprise Configuration

Based on this analysis, InFusion requires an enterprise with multiple divisions, ledgers, legal employers, payroll statutory units, tax reporting units, legislative data groups, departments, cost centers, and business units.

This figure illustrates the enterprise configuration that results from the analysis of InFusion Corporation.





FAQs for Define Initial Configuration

What happens if I don't use the Enterprise Structures Configurator to set up my enterprise structures?

The Enterprise Structures Configurator is an interview-based tool that guides you through setting up divisions, legal entities, business units, and reference data sets. If you do not use the Enterprise Structures Configurator, then you must set up your enterprise structure using the individual tasks that correspond to each enterprise component. In addition, you can't set up multiple configurations and compare different scenarios. Using the Enterprise Structures Configurator is the recommended process for setting up your enterprise structures.

What's an ultimate holding company?

The legal entity that represents the top level in your organization hierarchy, as defined by the legal name entered for the enterprise. This designation is used only to create an organization tree, with these levels:

- Ultimate holding company as the top level
- Divisions and country holding companies as the second level
- · Legal employers as the third level

Define Enterprise for Fusion Accounting Hub

Enterprise: Explained

An enterprise is a collection of legal entities under common control and management.

Enterprise Defined

When implementing Oracle Fusion Applications you operate within the context of an enterprise that has already been created in the application for you. This is either a predefined enterprise or an enterprise that has been created in the application by a system administrator. An enterprise organization captures the name of the deploying enterprise and the location of the headquarters. In Oracle Fusion Applications, an organization classified as an enterprise is defined before defining any other organizations in the HCM Common Organization Model. All other organizations are defined as belonging to an enterprise.

Managing Enterprise Information for Non-HCM Users: Explained

The Manage Enterprise HCM Information task includes default settings for your enterprise such as the employment model, worker number generation, and so on. If you are not implementing Oracle Fusion Human Capital Management (HCM), then



the only action you may need to perform using this task is to change the enterprise name, if necessary. The other settings are HCM-specific and are not relevant outside of Oracle Fusion HCM.

Define Legal Jurisdictions and Authorities

Jurisdictions and Legal Authorities: Explained

You are required to register your legal entities with legal authorities in the jurisdictions where you conduct business. Register your legal entities as required by local business requirements or other relevant laws. For example, register your legal entities for tax reporting to report sales taxes or value added taxes.

Define jurisdictions and related legal authorities to support multiple legal entity registrations, which are used by Oracle Fusion Tax and Oracle Fusion Payroll. When you create a legal entity, the Oracle Fusion Legal Entity Configurator automatically creates one legal reporting unit for that legal entity with a registration.

Jurisdictions: Explained

Jurisdiction is a physical territory such as a group of countries, country, state, country, or parish where a particular piece of legislation applies. French Labor Law, Singapore Transactions Tax Law, and US Income Tax Laws are examples of particular legislation that apply to legal entities operating in different countries' jurisdictions. Judicial authority may be exercised within a jurisdiction.

Types of jurisdictions are:

- · Identifying Jurisdiction
- Income Tax Jurisdiction
- Transaction Tax Jurisdiction

Identifying Jurisdiction

For each legal entity, select an identifying jurisdiction. An identifying jurisdiction is your first jurisdiction you must register with to be allowed to do business in a country. If there is more than one jurisdiction that a legal entity must register with to commence business, select one as the identifying jurisdiction. Typically the identifying jurisdiction is the one you use to uniquely identify your legal entity.

Income tax jurisdictions and transaction tax jurisdictions do not represent the same jurisdiction. Although in some countries, the two jurisdictions are defined at the same geopolitical level, such as a country, and share the same legal authority, they are two distinct jurisdictions.

Income Tax Jurisdiction

Create income tax jurisdictions to properly report and remit income taxes to the legal authority. Income tax jurisdictions by law impose taxes on your financial income generated by all your entities within their jurisdiction. Income tax is a key source of funding that the government uses to fund its activities and serve the public.



Transaction Tax Jurisdiction

Create transaction tax jurisdictions through Oracle Fusion Tax in a separate business flow, because of the specific needs and complexities of various taxes. Tax jurisdictions and their respective rates are provided by suppliers and require periodic maintenance. Use transaction tax jurisdiction for legal reporting of sales and value added taxes.

Legal Authorities: Explained

A legal authority is a government or legal body that is charged with powers to make laws, levy and collect fees and taxes, and remit financial appropriations for a given jurisdiction.

For example, the Internal Revenue Service is the authority for enforcing income tax laws in United States. In some countries, such as India and Brazil, you are required to print legal authority information on your tax reports. Legal authorities are defined in the Oracle Fusion Legal Entity Configurator. Tax authorities are a subset of legal authorities and are defined using the same setup flow.

Legal authorities are not mandatory in Oracle Fusion Human Capital Management (HCM), but are recommended and are generally referenced on statutory reports.

Creating Legal Jurisdictions, Addresses and Authorities: Examples

Define legal jurisdictions and related legal authorities to support multiple legal entity registrations, which are used by Oracle Fusion Tax and Oracle Fusion Payroll.

Legal Jurisdictions

Create a legal jurisdiction by following these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Jurisdictions > Go to Task.
- 2. Select Create.
- 3. Enter a unique **Name**, United States Income Tax.
- **4.** Select a **Territory**, United States.
- 5. Select a **Legislative Category**, Income tax.
- Select Identifying, Yes. Identifying indicates the first jurisdiction a legal entity must register with to do business in a country.
- 7. Enter a **Start Date** if desired. You can also add an **End Date** to indicate a date that the jurisdiction may no longer be used.
- 8. Select a **Legal Entity Registration Code**, EIN or TIN.
- 9. Select a Legal Reporting Unit Registration Code, Legal Reporting Unit Registration Number.
- 10. Optionally enter one or more Legal Functions.
- 11. Save and Close.

Legal Addresses for Legal Entities and Reporting Units

Create a legal address for legal entities and reporting units by following these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Address > Go to Task.
- 2. Select Create.
- 3. Select Country.
- 4. Enter Address Line 1, Oracle Parkway.



- 5. Optionally enter Address Line 2, and Address Line 3.
- 6. Enter or Select **Zip Code**, 94065.
- 7. Select Geography 94065 and Parent Geography Redwood Shores, San Mateo, CA.
- 8. Optionally enter a Time Zone, US Pacific Time.
- 9. OK.
- 10. Save and Close.

Legal Authorities

Create a legal authority by following these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Authorities > Go to Task.
- 2. Enter the **Name**, California Franchise Tax Board.
- 3. Enter the **Tax Authority Type**, Reporting.
 - Note: Create an address for the legal authority.
- 4. Select Create.
- 5. The **Site Number** is automatically assigned.
- 6. Optionally enter a Mail Stop.
- 7. Select Country, United States
- 8. Enter Address Line 1, 121 Spear Street, Suite 400.
- 9. Optionally enter Address Line 2, and Address Line 3.
- 10. Enter or Select **Zip Code**, 94105.
- 11. Select Geography 94105 and Parent Geography San Francisco, San Francisco, CA.
- 12. OK.
- **13.** Optionally enter a **Time Zone**, US Pacific Time.
- 14. Optionally click the One-Time Address check box.
- **15.** The **From Date** defaults to today's date. Update if necessary.
- 16. Optionally enter a **To Date** to indicate the last day the address can be used.
 - Note: You can optionally enter Address Purpose details.
- 17. Select Add Row.
- **18.** Select **Purpose**.
- **19.** The **Purpose from Date** will default to today's date.
- Optionally enter a Purpose to Date.
- 21. OK.
- 22. Save and Close.

Define Geographies

Define Geographies: Overview

Setting up your geography structure and master geographies correctly in the Trading Community Model is critical to the proper utilization and management of Oracle Enterprise Resource Planning (ERP) Cloud applications.



The geography structure and master geography data is shared across multiple product families and applications. Address validation ensures complete and valid master address data across all location entities across product applications. In addition, complete and valid master data is critical to ensure accurate transaction tax calculation.

You can either define your geography structure and corresponding master geographies manually or import these geography entities. Options include using the:

- 1. Manage Geographies page
- 2. Manage File Import Activities task

For more information about managing your geographies, see the Define Geographies section in the Define Enterprise Structures chapter in the Oracle Financials Cloud Implementing Common Features guide on Oracle Help Center (https://docs.oracle.com).

Manage Geographies

Use the Manage Geographies page to manually define your geography structure, hierarchy, mapping, and validation. Manually define your geographies when you have a simple geography requirement with a limited number of geographies within an individual country.

Manage File Import Activities

You can upload the geography structure and hierarchy using the Manage File Import Activities task. Once you upload your geography data successfully, you can use the Manage Geographies page to:

- Validate your data
- Enable geography validation
- Enable tax validation

Set the address validation for applicable countries to **Error** to ensure that no invalid or incomplete master address data is created for any location entities. The accuracy of master address data is critical to successful implementations.

Use data import when you have more complex geography requirements, such as for the United States and Canada.

Related Topics

- Geography Structure, Hierarchy, and Validation: How They Fit Together
- Managing Geography Structures, Hierarchies, and Validation: Worked Example
- Importing Geographies Using File-Based Import: Explained
- Oracle Financials Cloud Implementing Common Features

Define Legal Entities

Creating Legal Entities, Registrations, and Reporting Units: Examples

Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, and income, pay transaction taxes, or perform intercompany trading.



Legal Entity

Create a legal entity by following these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Entity > Go to Task.
- 2. Accept the default Country, United States.
- 3. Enter Name, InFusion USA West.
- 4. Enter Legal Entity Identifier, US0033.
- 5. Optionally enter **Start Date**. When the start date is blank the legal entity is effective from the creation date.
- 6. Optionally enter an End Date.
- Optionally, if your legal entity should be registered to report payroll tax and social insurance, select the Payroll statutory unit check box.
- 8. Optionally, if your legal entity has employees, select the Legal employer check box.
- 9. Optionally, if this legal entity is not a payroll statutory unit, select an existing payroll statutory unit to report payroll tax and social instance on behalf of this legal entity.
- 10. Enter the Registration Information
- 11. Accept the default **Identifying Jurisdiction**, United States Income Tax.
- 12. Search for and select a Legal Address, 500 Oracle Parkway, Redwood Shores, CA 94065.

The legal address must have been entered previously using the Manage Legal Address task.

- 13. OK
- **14.** Optionally enter a **Place of Registration**.
- 15. Enter the EIN or TIN.
- 16. Enter the Legal Reporting Unit Registration Number.
- 17. Save and Close.
- 18. Navigator > Setup and Maintenance > Define Legal Entries > Manage Legal Entity > Select... to set scope.
- 19. Select the Manage Legal Entity.
- 20. In the *Legal Entity list, select Select and Add.
- 21. Click Apply and Go to Task.
- 22. Select your legal entity.
- 23. Save and Close on the very bottom of the window.

This sets the scope for your task list to the selected legal entity.

24. Save and Close.

Legal Entity Registrations

A legal entity registration with the same name as that of the legal entity is created by default. To verify this, locate the **Manage Legal Entity Registrations** task and then select **Go to Task**. To create another registration for the legal entity follow these steps:

- 1. Navigator > Setup and Maintenance > Manage Legal Entity Registrations: Verify that the Legal Entity scope value is set correctly.
- 2. Go to Task.
- 3. Select Create.
- 4. Enter Jurisdiction.
- 5. Enter Registered Address.
- 6. Enter Registered Name.
- Optionally enter Alternate Name, Registration Number, Place of Registration, Issuing Legal Authority, and Issuing Legal Authority Address, Start Date, and End Date.



8. Save and Close.

Legal Reporting Unit

When a legal entity is created, a legal reporting unit with the same name as that of the entity is also automatically created. To create more legal reporting units or modify the settings follow these steps:

- 1. Navigator > Setup and Maintenance > Define Legal Reporting Unit. > Manage Legal Reporting Unit. Verify that the Legal Entity scope value is set correctly.
- 2. Go to Task
- 3. Select Create.
- 4. Enter Territory, United States.
- 5. Enter Name.
- **6.** Optionally enter a **Start Date**.
- 7. Enter Registration Information.
- **8.** Search for and select **Jurisdiction**.
- 9. Enter Main Legal Reporting Unit information.
- 10. Select the value Yes or No for the **Main Legal Reporting Unit**. Set value to yes only if you are creating a new main (primary) legal reporting unit.
- 11. Enter the Main Effective Start Date, 1/1/11.
- 12. Save and Close.

Related Topics

Planning Legal Reporting Units: Points to Consider

Legal Entities: Explained

A legal entity is a recognized party with rights and responsibilities given by legislation.

Legal entities have the following rights and responsibilities to:

- Own property
- Trade
- Repay debt
- Account for themselves to regulators, taxation authorities, and owners according to rules specified in the relevant legislation

Their rights and responsibilities may be enforced through the judicial system. Define a legal entity for each registered company or other entity recognized in law for which you want to record assets, liabilities, expenses and income, pay transaction taxes, or perform intercompany trading.

A legal entity has responsibility for elements of your enterprise for the following reasons:

- Facilitating local compliance
- Minimizing the enterprise's tax liability
- Preparing for acquisitions or disposals of parts of the enterprise
- Isolating one area of the business from risks in another area. For example, your enterprise develops property and
 also leases properties. You could operate the property development business as a separate legal entity to limit risk to
 your leasing business.



The Role of Your Legal Entities

In configuring your enterprise structure in Oracle Fusion Applications, the contracting party on any transaction is always the legal entity. Individual legal entities:

- Own the assets of the enterprise
- Record sales and pay taxes on those sales
- Make purchases and incur expenses
- Perform other transactions

Legal entities must comply with the regulations of jurisdictions, in which they register. Europe now allows for companies to register in one member country and do business in all member countries, and the US allows for companies to register in one state and do business in all states. To support local reporting requirements, legal reporting units are created and registered.

You are required to publish specific and periodic disclosures of your legal entities' operations based on different jurisdictions' requirements. Certain annual or more frequent accounting reports are referred to as statutory or external reporting. These reports must be filed with specified national and regulatory authorities. For example, in the United States (US), your publicly owned entities (corporations) are required to file quarterly and annual reports, as well as other periodic reports, with the Securities and Exchange Commission (SEC), which enforces statutory reporting requirements for public corporations.

Individual entities privately held or held by public companies do not have to file separately. In other countries, your individual entities do have to file in their own name, as well as at the public group level. Disclosure requirements are diverse. For example, your local entities may have to file locally to comply with local regulations in a local currency, as well as being included in your enterprise's reporting requirements in different currency.

A legal entity can represent all or part of your enterprise's management framework. For example, if you operate in a large country such as the United Kingdom or Germany, you might incorporate each division in the country as a separate legal entity. In a smaller country, for example Austria, you might use a single legal entity to host all of your business operations across divisions.

Creating Legal Entities in the Enterprise Structures Configurator: Points to Consider

Use the Enterprise Structures Configurator (ESC), to create legal entities for your enterprise automatically, based on the countries in which divisions of your business operate, or you can upload a list of legal entities from a spreadsheet.

Automatically Creating Legal Entities

If you are not certain of the number of legal entities that you need, you can create them automatically. To use this option, you first identify all of the countries in which your enterprise operates. The application opens the Map Divisions by Country page, which contains a matrix of the countries that you identified, your enterprise, and the divisions that you created. You select the check boxes where your enterprise and divisions intersect with the countries to identify the legal entities that you want the application to create. The enterprise is included for situations where your enterprise operates in a country, acts on behalf of several divisions within the enterprise, and is a legal employer in a country. If you select the enterprise for a country, the application creates a country holding company.

The application automatically creates the legal entities that you select, and identifies them as payroll statutory units and legal employers. For each country that you indicated that your enterprise operates in, and for each country that you created a location for, the application also automatically creates a legislative data group.

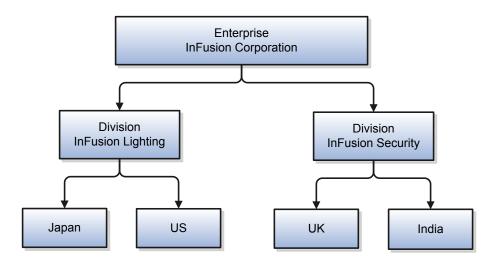


Any legal entities that you create automatically cannot be deleted from the Create Legal Entities page within the Enterprise Structures Configurator. You must return to the Map Divisions by Country page and deselect the legal entities that you no longer want.

Example: Creating Legal Entities Automatically

InFusion Corporation is using the ESC to set up its enterprise structure. The corporation has identified two divisions, one for Lighting, and one for Security. The Lighting division operates in Japan and the US, and the Security division operates in the UK and India.

This figure illustrates InFusion Corporation's enterprise structure.



This table represents the selections that InFusion Corporation makes when specifying which legal entities to create on the Map Divisions by Country page.

Country	Enterprise	InFusion Lighting	InFusion Security
Japan	No	Yes	No
US	No	Yes	No
UK	No	No	Yes
India	No	No	Yes

Based on the selections made in the preceding table, the ESC creates the following four legal entities:

- InFusion Lighting Japan LE
- InFusion Lighting US LE
- InFusion Security UK LE
- InFusion Security India LE



Creating Legal Entities Using a Spreadsheet

If you have a list of legal entities already defined for your enterprise, you can upload them from a spreadsheet. To use this option, you first download a spreadsheet template, then add your legal entity information to the spreadsheet, and then upload directly to your enterprise configuration. You can export and import the spreadsheet multiple times to accommodate revisions.

Related Topics

- HCM Organization Models: Examples
- Payroll Statutory Units, Legal Employers, and Tax Reporting Units: How They Work Together
- Using Desktop Integrated Excel Workbooks: Points to Consider

Legal Entity in Oracle Fusion: Points to Consider

Oracle Fusion Applications support the modeling of your legal entities. If you make purchases from or sell to other legal entities, define these other legal entities in your customer and supplier registers. These registers are part of the Oracle Fusion Trading Community Architecture.

When your legal entities are trading with each other, represent them as legal entities and as customers and suppliers in your customer and supplier registers. Use legal entity relationships to determine which transactions are intercompany and require intercompany accounting. Your legal entities can be identified as legal employers and therefore, are available for use in Human Capital Management (HCM) applications.

Several decisions you should consider when you create legal entities.

- The importance of using legal entity on transactions
- Legal entity and its relationship to business units
- Legal entity and its relationship to divisions
- Legal entity and its relationship to ledgers
- Legal entity and its relationship to balancing segments
- · Legal entity and its relationship to consolidation rules
- Legal entity and its relationship to intercompany transactions
- Legal entity and its relationship to worker assignments and legal employer
- · Legal entity and payroll reporting
- · Legal reporting units

The Importance of Using Legal Entities on Transactions

All of the assets of the enterprise are owned by individual legal entities. Oracle Fusion Financials allow your users to enter legal entities on transactions that represent a movement in value or obligation.

For example, a sales order creates an obligation on the legal entity that books the order to deliver the goods on the acknowledged date. The creation also creates an obligation on the purchaser to receive and pay for those goods. Under contract law in most countries, damages can be sought for both:

- Actual losses, putting the injured party in the same state as if they had not entered into the contract.
- What is called loss of bargain, or the profit that would have made on a transaction.



In another example, if you revalued your inventory in a warehouse to account for raw material price increases, the revaluation and revaluation reserves must be reflected in your legal entity's accounts. In Oracle Fusion Applications, your inventory within an inventory organization is managed by a single business unit and belongs to one legal entity.

Legal Entity and Its Relationship to Business Units

A business unit can process transactions on behalf of many legal entities. Frequently, a business unit is part of a single legal entity. In most cases, the legal entity is explicit on your transactions. For example, a payables invoice has an explicit legal entity field. Your accounts payables department can process supplier invoices on behalf of one or many business units.

In some cases, your legal entity is inferred from your business unit that is processing the transaction. For example, Business Unit ACM UK has a default legal entity of InFusion UK Ltd. When a purchase order is placed in ACM UK, the legal entity InFusion UK Ltd is legally obligated to the supplier. Oracle Fusion Procurement, Oracle Fusion Project Portfolio Management, and Oracle Fusion Supply Chain applications rely on deriving the legal entity information from the business unit.<

Legal Entity and Its Relationship to Divisions

The division is an area of management responsibility that can correspond to a collection of legal entities. If wanted, you can aggregate the results for your divisions by legal entity or by combining parts of other legal entities. Define date-effective hierarchies for your cost center or legal entity segment in your chart of accounts to facilitate the aggregation and reporting by division. Divisions and legal entities are independent concepts.

Legal Entity and Its Relationship to Ledgers

One of your major responsibilities is to file financial statements for your legal entities. Map legal entities to specific ledgers using the Oracle Fusion General Ledger Accounting Configuration Manager. Within a ledger, you can optionally map a legal entity to one or more balancing segment values.

Legal Entity and Its Relationship to Balancing Segments

Oracle Fusion General Ledger supports up to three balancing segments. Best practices recommend one segment represents your legal entity to ease your requirement to account for your operations to regulatory agencies, tax authorities, and investors. Accounting for your operations means you must produce a balanced trial balance sheet by legal entity. If you account for many legal entities in a single ledger, you must:

- 1. Identify the legal entities within the ledger.
- 2. Balance transactions that cross legal entity boundaries through intercompany transactions.
- 3. Decide which balancing segments correspond to each legal entity and assign them in Oracle Fusion General Ledger Accounting Configuration Manager. Once you assign one balancing segment value in a ledger, then all your balancing segment values must be assigned. This recommended best practice facilitates reporting on assets, liabilities, and income by legal entity.

Represent your legal entities by at least one balancing segment value. You may represent it by two or three balancing segment values if more granular reporting is required. For example, if your legal entity operates in multiple jurisdictions in Europe, you might define balancing segment values and map them to legal reporting units. You can represent a legal entity with more than one balancing segment value. Do not use a single balancing segment value to represent more than one legal entity.

In Oracle Fusion General Ledger, there are three balancing segments. You can use separate balancing segments to represent your divisions or strategic business units to enable management reporting at the balance sheet level for each. This solution is used to empower your business unit and divisional managers to track and assume responsibility for their asset utilization or return on investment. Using multiple balancing segments is also useful when you know at the time of implementation that you are disposing of a part of a legal entity and want to isolate the assets and liabilities for that entity.

Implementing multiple balancing segments requires every journal entry that is not balanced by division or business unit, to generate balancing lines. You cannot change to multiple balancing segments after you begin using the ledger because your historical data is not balanced by the new balancing segments. Restating historical data must be done at that point.



If your enterprise regularly spins off businesses or holds managers accountable for utilization of assets, identify the business with a balancing segment value. If you account for each legal entity in a separate ledger, no requirement exists to identify the legal entity with a balancing segment value.

While transactions that cross balancing segments don't necessarily cross legal entity boundaries, all transactions that cross legal entity boundaries must cross balancing segments. If you make an acquisition or are preparing to dispose of a portion of your enterprise, you may want to account for that part of the enterprise in its own balancing segment even if the portion is not a separate legal entity. If you do not map legal entities sharing the same ledger to balancing segments, you cannot distinguish them using intercompany functionality or track individual equity.

Legal Entity and Its Relationship to Consolidation Rules

In Oracle Fusion Applications you can map legal entities to balancing segments and then define consolidation rules using your balancing segments. You are creating a relationship between the definition of your legal entities and their role in your consolidation.

Legal Entity and Its Relationship to Intercompany Transactions

Use Oracle Fusion Intercompany feature to create intercompany entries automatically across your balancing segments. Intercompany processing updates legal ownership within the enterprise's groups of legal entities. Invoices or journals are created as needed. To limit the number of trading pairs for your enterprise, set up intercompany organizations and assign then to your authorized legal entities. Define processing options and intercompany accounts to use when creating intercompany transactions and to assist in consolidation elimination entries. These accounts are derived and automatically entered on your intercompany transactions based on legal entities assigned to your intercompany organizations.

Intracompany trading, in which legal ownership isn't changed but other organizational responsibilities are, is also supported. For example, you can track assets and liabilities that move between your departments within your legal entities by creating departmental level intercompany organizations.



💡 Tip: In the Oracle Fusion Supply Chain applications, you can model intercompany relationships using business units, from which legal entities are derived.

Legal Entity and Its Relationship to Worker Assignments and Legal Employer

Legal entities that employ people are called legal employers in the Oracle Fusion Legal Entity Configurator. You must enter legal employers on worker assignments in Oracle Fusion HCM.

Legal Entity and Payroll Reporting

Your legal entities are required to pay payroll tax and social insurance such as social security on your payroll. In Oracle Fusion Applications, you can register payroll statutory units to pay and report on payroll tax and social insurance for your legal entities. As the legal employer, you might be required to pay payroll tax, not only at the national level, but also at the local level. You meet this obligation by establishing your legal entity as a place of work within the jurisdiction of a local authority. Set up legal reporting units to represent the part of your enterprise with a specific legal reporting obligation. You can also mark these legal reporting units as tax reporting units, if the legal entity must pay taxes as a result of establishing a place of business within the jurisdiction.



3 Define Financial Reporting Structures

Representing Your Enterprise Structure in Your Financial Reporting Structure: Overview

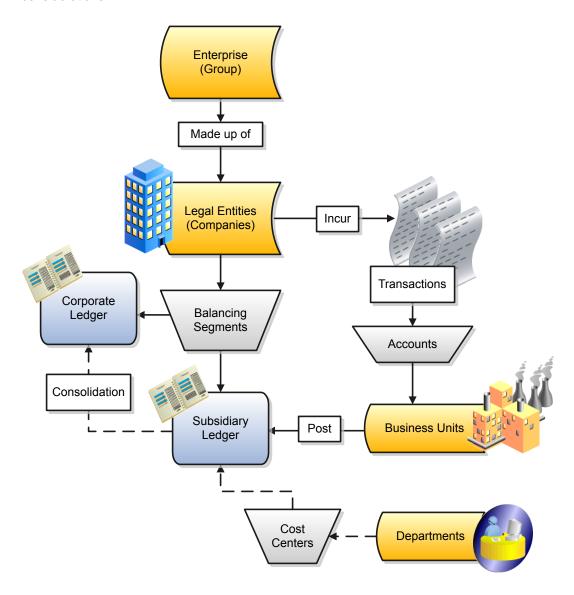
Represent your enterprise structures in your chart of accounts to track and report on your financial objectives and meet your reporting requirements. The benefit of representing your enterprise in the chart of accounts is the functionality which includes multidimensional reporting with its Essbase tool. Segments included in the chart of accounts become dimensions in Essbase. The recorded data is automatically loaded into the Essbase cube when you post your journal entries. The Essbase tool includes powerful functionality for analysis and reporting on your financial data.

Financial Enterprise Structure: How It Fits Together

Oracle Fusion Applications are an integrated suite of business applications that connects and automates the entire flow of the business process across both front and back-office operations and addresses the needs of a global enterprise. The process of designing the enterprise structure, including the accounting configuration, is the starting point for an implementation. This



process often includes determining financial, legal, and management reporting requirements and examining consolidation considerations.



Accounting Configuration Components

The accounting configuration components are:

- Ledgers: The main record-keeping structure. A ledger records the transactional balances by defining a chart of
 accounts with a calendar and currency, and accounting rules, implemented in an accounting method. The ledger is
 associated with the subledger transactions for the business units that are assigned to it, and provides context and
 accounting for them.
- Balancing Segments: Use these chart of accounts element to represent and track both legal and management entities. The primary balancing segment is used to represents your legal entities. The Second and Third balancing segments are used to implement management reporting and analysis. Balancing segments provide automatic



balancing functionality by legal entity for journal entries, including intercompany and intracompany entries, suspense posting, and rounding imbalances.

- Cost Centers: The component that aggregates elements of natural expenses to identify functional costs. A cost
 center can be the smallest segment of an organization for which costs are collected and reported. Not all cost
 centers represent organizations. A manager is assigned responsibility for cost control and is assigned both a
 department and a cost center; in which case the cost center and department might be identified with each other.
 However, a finance department manager might have separate cost centers for finance cost and audit costs and a
 Research and Development department manager might have separate cost centers for research and development.
 - Cost centers are represented by segment values in the chart of accounts that indicate the functional areas of your business, such as accounting, facilities, shipping, or human resources. You might keep track of functional areas at a detailed level, but produce summary reports that group cost centers into one or more departments. Cost center values are also used by Oracle Fusion Assets to assist the managers in tracking locations and accounting for assets assigned to their departments.
- Accounts: The code in the chart of accounts that uniquely identifies each type of transactions recorded in the ledger
 and subledgers. The account segment is mapped to a dimension in the GL Balances Cube to enable reporting and
 inquiry. This functionality uses Oracle Fusion Business Intelligence to analyze and drill into expense and revenue
 transactions.

Representing Legal Entities, Business Units, Departments in Chart of Accounts

The following list provides information about how to represent legal entities, business units, and departments in chart of accounts.

- Representing Legal Entities in the Chart of Accounts: Legal entity is the term used in Oracle Fusion Applications for registered companies and other organizations recognized in law as having a legal existence and as parties with given rights and responsibilities.
 - Legal entities are created in the applications and then assigned balancing segment values, sometimes called company codes in your ledgers during accounting configuration.
- Representing Business Units in the Chart of Accounts: A business unit (BU) is part of an enterprise managed by a manager with profit and loss responsibility. The business unit is generally tracked in the chart of accounts. A business unit can be represented by a single ledger. For example, in countries where you need document sequencing for unique transaction sequencing within a legal entity, you can have a single ledger with a single business unit and legal entity.

A business unit can also be identified in the chart of accounts as a:

- Management segment value
- Balancing segment value
- Roll up of cost center segments using hierarchies

For example, a business unit manager is responsible for working capital utilization or overall asset utilization. You identify the business unit as a balancing segment value, to enable calculation of ratios for various utilization measurements.

A business unit is assigned to a primary ledger, as well as a default legal entity when the business unit is configured. A BU identifies the primary ledger that subledger transactions are posted to, facilitating the use of more than one BU



per general ledger. Each business unit posts transactions to a single primary ledger. For example, a shared service center handles all the procurement functions for the entire company. The procurement transactions are posted to the business unit's ledger with intercompany entries to other ledgers as needed.

Representing Departments in the Chart of Accounts: A department is an organizational structure with one or more
operational objectives or responsibilities that exist independently of its manager and that has one or more employees
assigned to it. The manager of a department is typically responsible for business deliverables, personnel resource
management, competency management, and occasionally, for cost control and asset tracking.

In Oracle Fusion Applications, departments can be set up in Oracle Fusion Human Capital Management (HCM). If wanted, they can also be represented by a unique segment in the chart of accounts or a group of cost center values.

Modeling Your Financial Reporting Structure in Oracle Fusion: Example

This example uses a fictitious global company to demonstrate the analysis that can occur during the financial reporting structure planning process.

Scenario

Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion Enterprise Resource Planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers. You are chairing a committee to discuss creation of a model for your global financial reporting structure including your chart of accounts for both your US and UK operations.

InFusion Corporation

InFusion Corporation has 400 plus employees and revenue of 120 million US dollars. Your product line includes all the components to build and maintain air quality monitoring systems for homes and businesses. You have two distribution centers and three warehouses that share a common item master in the US and UK. Your financial services organization provides funding to your customers for the start-up costs of these systems.

Analysis

The following are elements you must consider in creating your model for your financial reporting structure.

- Your company is required to report using US Generally Accepted Accounting Principles (GAAP) standards and UK Statements of Standard Accounting Practice and Financial Reporting Standards. How many ledgers do you plan to have to achieve proper statutory reporting?
- Your financial services line of business has a different year end. Do you need a different calendar? Your financial services entity must report with average balance sheets. This feature of Oracle Fusion General Ledger provides you



with the ability to track average and end-of-day balances, report average balance sheets, and create custom reports using both standard and average balances.

- Your corporate management requires reports showing total organizational performance with drill-down capability to the supporting details. Do you need multiple balancing segment hierarchies to achieve proper rollup of balances for reporting requirements?
- Legal entity balancing options: Do you produce financial statements by one or more than one legal entity? Can you
 record multiple legal entities in one ledger or do you require multiple ledgers? Are you upgrading to Oracle Fusion
 Applications or a new install? If an upgrade, is your current financial reporting structure meeting your reporting
 needs?

Global Financial Reporting Model

The following figure and table summarize the model that your committee has designed and uses number values to provide a sample representation of your financial reporting structure. The model includes the following recommendations:

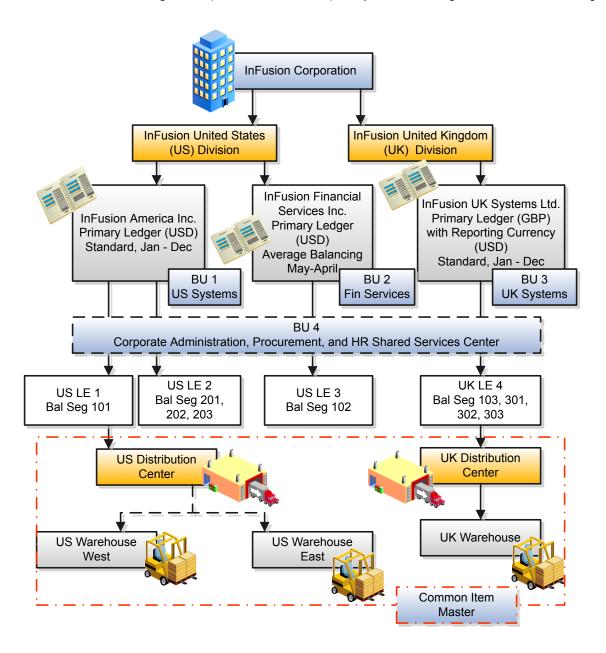
- Creation of three separate ledgers representing your separate legal entities:
 - o InFusion America Inc.
 - InFusion Financial Services Inc.
 - InFusion UK Services Ltd.
- Data security is controlled by balancing segment values using Oracle Fusion General Ledger data access sets

Recommendations for the chart of accounts design include:

• Segments required for cost centers with hierarchical rollups to departments providing reporting at both the detail (cost center) and summary (department) level.



Accounts configured to provide drill down capability to the subledger transactions, enabling analysis of data.



Decision	InFusion America, Inc.	InFusion Financial Services, Inc.	InFusion UK Systems, Ltd.
Type of Ledgers	Primary	Primary	Primary with the use of Reporting Currency functionality
Legal Entity Codes	US Legal Entity 1: US Corporate US Legal Entity 2: US Systems	US Legal Entity 3: US Financial Services	UK Legal Entity 4: UK Systems
Balancing Segments	101: US Corporate	102: US Financial Services	103: UK Systems



Decision	InFusion America, Inc.	InFusion Financial Services, Inc.	InFusion UK Systems, Ltd.
	201: US Systems Components202: US Systems Installations203: US Systems Maintenance		301: Components 302: UK Systems Installations 303: UK Systems Maintenance
Currencies for Reporting	US Dollar (USD)	US Dollar (USD)	Great Britain Pounds Sterling (GBP) US Dollar (USD)
Calendar Ending Date	December 31st	April 30th	December 31st
Business Units (BU)*	BU 1: US Systems BU 4: Corporate (Shared Service Center)	BU 2: Financial Services	BU 3: UK Systems
Balances Storage Method	Standard Balances	Average and Standard Balances	Standard Balances
Locations represented by cost centers in the chart of accounts.	Headquarters US Distribution Center (BU 1) US Warehouse West US Warehouse East	Headquarters	UK Distribution Center (BU 3) UK Warehouse

Note: In the chart of accounts, cost centers, with hierarchical rollups, represent business units. InFusion Corporation is also a legal entity but is not discussed in this example.

Manage Chart of Accounts

Chart of Accounts: Explained

The chart of accounts is the underlying structure for organizing financial information and reporting. An entity records transactions with a set of codes representing balances by type, expenses by function, and other divisional or organizational codes that are important to its business.

A well-designed chart of accounts provides the following benefits:

- Effectively manages an organization's financial business.
- Supports the audit and control of financial transactions.
- Provides flexibility for management reporting and analysis.
- Anticipates growth and maintenance needs as organizational changes occur.
- Facilitates an efficient data processing flow.



- Enables delegation of responsibility for cost control, profit attainment, and asset utilization.
- Measures performance against corporate objectives by your managers.

The chart of accounts facilitates aggregating data from different operations, from within an operation, and from different business flows, thus enabling the organization to report using consistent definitions to their stakeholders in compliance with legislative and corporate reporting standards and aiding in management decisions.

Best practices include starting the design from external and management reporting requirements and making decisions about data storage in the general ledger, including thick versus thin general ledger concepts.

Chart of Accounts: How Its Components Fit Together

The important elements in a basic chart of accounts in Oracle Fusion Applications included a structure that defines the account values, segments and their labels, and rules (security and validation). Account combinations link the values in the segments together and provide the accounting mechanism to capture financial transactions.

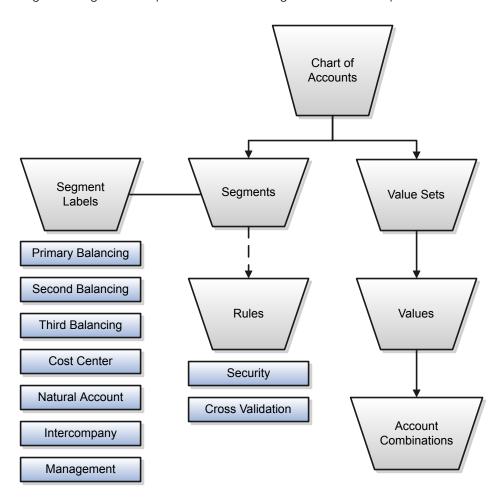




Chart of Accounts

The chart of accounts defines the number and attributes of various segments, including:

- Order of segments
- Width of segments
- Prompts
- Segment labels, such as balancing, natural account, and cost center.

The chart of accounts further defines:

- Combination of value sets associated with each segment
- Type of segment
- Default values for the segments
- Additional conditions designating the source of the values using database tables
- Required and displayed properties for the segments

Segments

A chart of accounts segment is a component of the account combination. Each segment has a value set attached to it to provide formatting and validation of the set of values used with that segment. The combination of segments creates the account combination used for recording and reporting financial transactions. Examples of segments that may be found in a chart of accounts are company, cost center, department, division, region, account, product, program, and location.

Value Sets and Values

The value sets define the attributes and values associated with a segment of the chart of accounts. You can think of a value set as a container for your values. You can set up your flexfield so that it automatically validates the segment values that you enter against a table of valid values. If you enter an invalid segment value, a list of valid values appears automatically so that you can select a valid value. You can assign a single value set to more than one segment, and you can share value sets across different flexfields.



Caution: You must use Independent validation only for the Accounting Key Flexfield value sets. Other validations prevent you from using the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration. Dependent values sets are not supported.

Segment Labels

Segment labels identify certain segments in your chart of accounts and assign special functionality to those segments. Segment labels were referred to as flexfield qualifiers in Oracle E-Business Suite. Here are the segment labels that are available to use with the chart of accounts.

- Balancing: Ensures that all journals balance for each balancing segment value or combination of multiple balancing segment values to use in trial balance reporting. The three balancing segment labels are: primary, second, and third balancing. The primary balancing segment label is required.
- Cost Center: Facilitates grouping of natural accounts by functional cost types, accommodating tracking of specific business expenses across natural accounts. As cost centers combine expenses and headcount data into costs, they are useful for detailed analysis and reporting. Cost centers are optional, but required if you are accounting for depreciation, additions, and other transactions in Oracle Fusion Assets, and for storing expense approval limits in Oracle Fusion Expense Management. If you are implementing Oracle Fusion Procurement, you can use cost centers for business intelligence reporting and to route transactions for approval.



- **Natural Account**: Determines the account type (asset, liability, expense, revenue, or equity) and other information specific to the segment value. The natural account segment label is required.
- **Management**: Optionally, denotes the segment that has management responsibility, such as the department, cost center, or line of business. Also can be attached to the same segment as one of the balancing segments to make legal entity reporting more granular.
- Intercompany: Optionally, assigns the segment to be used in intercompany balancing functionality.
- Note: All segments have a segment qualifier that enables posting for each value. The predefined setting is Yes to post.

Account Combinations

An account combination is a completed code of segment values that uniquely identifies an account in the chart of accounts, for example 01-2900-500-123, might represent InFusion America (company)-Monitor Sales (division)-Revenue (account)-Air Filters (product).

Rules

The chart of accounts uses two different types of rules to control functionality.

- **Security rules**: Prohibit certain users from accessing specific segment values. For example, you can create a security rule that grants a user access only to his or her department.
- Cross-validation rules: Control the account combinations that can be created during data entry. For example, you may decide that sales cost centers 600 to 699 should enter amounts only to product sales accounts 4000 to 4999.

Essbase Character and Word Limitations

The following is a comprehensive list of character and word limitations that apply to Essbase. All of the limitations apply to all of the Oracle Fusion General Ledger configurations summarized in the table.

Oracle Fusion General Ledger Configuration	Maps to Essbase
Chart of Account Name	Cube Name
Chart of Account Segment Name	Dimension Name
Chart of Accounts Segment Value	Dimension Member Name
Chart of Accounts Segment Value Description	Alias for Member
Tree and Tree Version Name	Dimension Member Name
Primary Ledger Name	Dimension Member Name in Ledger Dimension
Secondary Ledger Name	Dimension Member Name in Ledger Dimension



Oracle Fusion General Ledger Configuration	Maps to Essbase
Reporting Currency Name	Dimension Member Name in Ledger Dimension
Ledger Set Name	Dimension Member Name in Ledger Dimension
Accounting Calendar Period Names	Dimension Member Name in Accounting Period Name
Scenario Name Defined in Predefined Value Set Called Accounting Scenario	Dimension Member Name in Scenario Dimension

Even if case sensitivity is enabled in an aggregate storage outline for which duplicate member names is enabled, do not use matching dimension names with only case differences. For example, do not:

- Name two dimensions Product and product.
- Use quotation marks or brackets.
- Use tabs in dimension, member, or alias names.
- · Use accent characters.
- Use the characters for dimension or member names.

Restricted Characters

The following is a list of characters that are restricted and cannot be used at the beginning of dimension, member, or alias names.

Character	Meaning
@	at sign
\	backslash
,	comma
-	dash, hyphen, or minus sign
	For the accounting calendar period names, you can use a hyphen or an underscore in the middle of an accounting calendar period name. For example: Jan-15 or Adj_Dec-15 can be used successfully.
=	equal sign
<	less than sign
0	parentheses
	period
+	plus sign
	period



Character	Meaning
1	single quotation mark
-	underscore
	For the accounting calendar period names, you can use a hyphen or an underscore in the middle of an accounting calendar period name. For example: Jan-15 or Adj_Dec-15 can be used successfully.
I	vertical bar

Other Restrictions

- Don't place spaces at the beginning or end of names. Essbase ignores such spaces.
- Don't use these types of words as dimension or member names:
 - o Calculation script commands, operators, and keywords.
 - Report writer commands.
 - Function names and function arguments.
 - o Names of other dimensions and members (unless the member is shared).
 - o Generation names, level names, and aliases in the database.
 - Any of these words in the table below:

List 1	List 2	List 3
ALL	AND	ASSIGN
AVERAGE	CALC	CALCMBR
COPYFORWARD	CROSSDIM	CURMBRNAME
DIM	DIMNAME	DIV
DYNAMIC	EMPTYPARM	EQ
EQOP	EXCEPT	EXP
EXPERROR	FLOAT	FUNCTION
GE	GEN	GENRANGE
GROUP	GT	ID
IDERROR	INTEGER	LE
LEVELRANGE	LOOPBLOCK	LOOPPARMS



List 1	List 2	List 3
LT	MBR	MBRNAME
MBRONLY	MINUS	MISSING, #MISSING
MUL	MULOP	NE
NON	NONINPUT	NOT
OR	PAREN	PARENPARM
PERCENT	PLUS	RELOP
SET	SKIPBOTH	SKIPMISSING
SKIPNONE	SKIPZERO	ТО
TOLOCALRATE	TRAILMISSING	TRAILSUM
UMINUS	UPPER	VARORXMBR
XMRONLY	\$\$\$UNIVERSE\$\$\$	#MI

Chart of Accounts Structure and Instances: Critical Choices

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created. Consider the critical choices list here when creating your hart of accounts instances and structures.

Chart of Accounts Structure

When creating the segments for the chart of accounts structure, you must enter segment sequence numbers that are consecutive and begin with the number one.

Chart of Accounts Structure Instance

For segments in your chart of account structure instance that you expect to contain a large number of distinct values, you must perform the following steps:

- In the chart of accounts definition, mark the segment query required option as selectively required. To perform searches in the transactional user interface, you have to specify the following segment as a mandatory search criteria.
- Create required indexes in the GL_CODE_COMBINATIONS table for segments that are selectively required.



Caution: For the Accounting Key Flexfield value sets used in your chart of accounts structure and instance, you must use independent validation only. If you use other validations, you may not be able to use the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration.

Creating Chart of Accounts Structure and Instances: Examples

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created. A chart of accounts structure defines the key attributes for your chart of accounts, such as the number of segments, the segment seguences, the segment names, segment prompts, segment labels, for example natural account and primary balancing, and default value sets.

The chart of accounts instance is exposed in the user interfaces and processes. By default, a chart of accounts instance inherits all the attributes of the chart of accounts structure, meaning that all instances of the same structure share a common shape and have the same segments in the same order. However, at the chart of accounts instance level, you can override the default value set assignments for your segments and assign a unique account hierarchy that determines the parent and child relationships between the value set values. At the chart of accounts instance level, determine if allow dynamic insertion is enabled to generate new account combinations dynamically instead of creating them manually.

Chart of Account Structure

You are creating a chart of accounts structure as you setup your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

- Navigator > Setup and Maintenance > Manage Chart of Accounts > Go To Task.
- 2. Select **General Ledger** from the Module list of values and click **Search**.
- 3. Click Manage Structures to open the Manage Key Flexfield Structures page.
- 4. Select the **General Ledger** row and click the **Create** to open the **Create Key Flexfield Structure** page.
- Enter a unique Structure Code, INFUSION_AM_COA_STRUCTURE, and Name, InFusion America COA Structure. Provide an optional Description, InFusion America Inc. Chart of Accounts Structure.
- 6. Select the Delimiter to visually separate your segment values.
- 7. Save.
- 8. To create a new segment, click the Create to open the Create Key Flexfield Segment page.
 - **a.** Enter the following parameters:

Parameter	Value
Segment Code	INFUSION_AM_CO
Name	InFusion America Company
Description	InFusion America Inc.
Sequence Number	1
Prompt	Company
Short Prompt	СО
Display Width	2



Parameter	Value
Column Name	Segment1
Default Value Set Code	INFUSION_ AM_COMPANY

- b. Select a segment label, **Primary Balancing Segment**, to indicate its purpose within your chart of accounts.
 - Note: Two segment labels are required: primary balancing segment and natural account segment. These labels are not used with each other or with other labels in a specific segment.
- c. Save and Close.
- d. Done.
- **e.** Define additional segments following the same process.

Chart of Account Instance

You are creating a chart of accounts instance as you setup your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

- 1. Navigator > Setup and Maintenance > Manage Chart of Accounts > Go To Task.
- 2. Select General Ledger from the Module list of values and click Search.
- 3. Select the General Ledger row and click Manage Structure Instances to open the Manage Key Flexfield Structure Instance page.
- 4. Click the Create icon to open the Create Key Flexfield Structure Instance page.
- 5. Enter a unique Structure Instance Code, INFUSION_AM_COA_INSTANCE, and Name, InFusion America COA Instance. Provide an optional Description, InFusion America Inc. Chart of Accounts Structure Instance.
- **6.** Select **Dynamic combination creation allowed** to indicate that you want to dynamically generate account combinations.
- **7.** Associate your instance with your Structure Name, **InFusion America Structure**.
 - Note: By default, an instance inherits the key attributes of the associated structure. Some attributes, such as the value set assigned to each the segment, can be modified.
- 8. Save.
- 9. Optionally, select the segment row and click **Edit** to modify instance segments.
- 10. Check Required, Displayed, and BI enabled check boxes.
 - Note: Check the Required and Displayed options for all segments including those intended for future use. The recommended best practice is to define one segment for future use and set a default value. This ensures room for expansion in your chart of accounts and that the extra segment is populated in the account combinations. Check the BI (Business Intelligence) enabled option to use key flexfield segments in Oracle Fusion Transactional Business Intelligence. The business intelligence check box is only valid when enabled on segments with segment labels. The second step is to populate the BI Object Name field for each of the segment labels in the Manage Segment Label page opened from the Manage Key Flexfields page.
- 11. OK.
- 12. Save and Close.



- **13.** Define additional instances following the same process.
 - Note: Alternatively, proceed directly with creating your value set values by selecting the corresponding Value Set Code in the Segment Instances table.
- 14. Done.
- 15. Deploy Flexfield.
- 16. OK.

Related Topics

• Enabling Key Flexfield Segments for Business Intelligence: Points to Consider

Creating One Chart of Accounts Structure with Many Instances: Example

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created.

Scenario

Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion enterprise resource planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers. You are chairing a committee to discuss creation of a model for your global financial reporting structure including your charts of accounts for both your US and UK operations.

InFusion Corporation

InFusion Corporation has 400 plus employees and revenue of \$120 million. Your product line includes all the components to build and maintain air quality monitoring (AQM) systems for homes and businesses.

Analysis

In Oracle Fusion General Ledger, the chart of accounts model is framed around the concept of a chart of accounts structure, under which one or more chart of accounts structure instances can be created.

Chart of Accounts Model

The chart of accounts structure provides the general outline of the chart of accounts and determines the number of segments, the type, the length, and the label (qualifier) of each segment. This forms the foundation of the chart of accounts definition object.

For each chart of accounts structure, it is possible to associate one or more chart of accounts structure instances. Chart of accounts structure instances under the same structure share a common configuration with the same segments, in the same order, and the same characteristics. Using one chart of accounts structure with multiple instances simplifies your accounting and reporting.

At the chart of accounts structure instance level, each segment is associated with a value set that conforms to the characteristic of that segment. For example, you assign a value set with the same segment type and length to each segment. You are using hierarchies with your chart of accounts segments. Each structure instance segment is assigned a tree code to indicate the source of the hierarchy information for the associated value set. The same value set can be used multiple



times within the same or across different chart of accounts instances within the same structure or in different structures. This functionality reduces your segment value creation and maintenance across your charts of accounts.

The collective assignment of value sets to each of the segments forms one chart of accounts instance. At the chart of accounts structure instance level, you can select to enable dynamic insertion. Dynamic insertion allows the creation of account code combinations automatically the first time your users enter that new account combination. The alternative is to create them manually. By deciding to enable dynamic insertion, you save data entry time and prevent delays caused by the manual creation of new code combinations. Well defined cross validation rules help prevent the creation of inappropriate account code combinations.

Perform deployment after a new chart of accounts structure and structure instances are defined or any of their modifiable attributes are updated. Deployment validates and regenerates the necessary objects to enable your charts of accounts and chart of accounts structure instances. By unifying and standardizing you organization's chart of accounts, you are positioned to take full advantage of future functionality in Oracle Fusion General Ledger.

In summary, you are recommending to your company to unify the organization's chart of accounts in a single chart of accounts structure based on chart of accounts commonalities across ledgers. You have also decided to use the chart of accounts structure instance construct to serve different accounting and reporting requirements by using value sets specific to each of your entities.

Configuring Chart of Account Segment for Business Intelligence: Explained

To map the Oracle Fusion General Ledger Accounting Flexfield in Oracle Fusion Transaction Business Intelligence (BI) Repository file (RPD) for Oracle Fusion Financials, populate values in the Manage Key Flexfields user interface. These values enable the chart of accounts segments for Oracle Fusion Transactional BI. The values also provide the mapping with BI Object names that are used as dimensions for each of the chart of accounts segments.

Check each of the Chart of Accounts segments' BI enabled check box on all segments that you intend to map in the RPD:

- 1. From your implementation project or the **Setup and Maintenance** page, query for **Manage Key Flexfields** > **Go to Task**.
- 2. Enter GL# in the **Key Flexfield Code**.
- 3. Search.
- 4. Click on Manage Structure Instances.
- 5. Search.
- 6. Click on the specific chart of accounts and Edit icon.
- 7. Click on the specific segment and the Edit icon.
- 8. Check the BI enabled check box.
- Save. This should be done for all segments in every Chart of Accounts Structure Instance that you intend to be mapped in the RPD.
- 10. Save and Close > Done.

Populate the **BI Object Name** for each of the **Segment Labels**. This name is the logical table name in the RPD which would be used as the dimension for the corresponding segment. Perform the following steps:

- From your implementation project or the Setup and Maintenance page, query for Manage Key Flexfields > Go to Task.
- 2. Enter GL# in the **Key Flexfield Code**.
- 3. Query for GL# as **Key Flexfield Code** in **Manage Key Flexfields** page.
- 4. Search.
- 5. Actions menu and click on Manage Segment Labels.
- 6. Populate the **BI Object Name** for all the segment labels that must be mapped in the RPD.



Segment Label Code	Bl Object Name
FA_COST_CTR	Dim - Cost Center
GL_BALANCING	Dim - Balancing Segment
GL_ACCOUNT	Dim - Natural Account Segment

7. Save.

- Note: For all the nonqualified segment labels, the **BI Object Name** should be populated with one of the following:
- Dim GL Segment1
- Dim GL Segment2
- Dim GL Segment3
- Dim GL Segment4
- Dim GL Segment5
- Dim GL Segment6
- Dim GL Segment7
- Dim GL Segment8
- Dim GL Segment9
- Dim GL Segment10

Deploy the flexfield using the **Deploy Flexfield** button from **Manage Key Flexfields** page. For more information about extending both key and descriptive flexfields into Oracle Fusion Transactional BI, refer to Oracle Fusion Transactional Business Intelligence Administrator's Guide.

Cost Centers and Departments: Explained

The two important components to be considered in designing your enterprise structure are cost centers and departments.

A cost center represents the smallest segment of an organization for which you collect and report costs. A department is an organization with one or more operational objectives or responsibilities that exist independently of its manager and has one or more workers assigned to it.

Cost Centers

A cost center represents the destination or function of an expense rather than the nature of the expense which is represented by the natural account. For example, a sales cost center indicates that the expense goes to the sales department.

A cost center is generally attached to a single legal entity. To identify the cost centers within a chart of accounts structure use one of these two methods:

Assign a cost center value in the value set for each cost center. For example, assign cost center values of PL04 and
G3J1 to your manufacturing teams in the US and India. These unique cost center values allow easy aggregation of
cost centers in hierarchies (trees) even if the cost centers are in different ledgers. However, this approach requires
defining more cost center values.



• Assign a balancing segment value with a standardized cost center value to create a combination of segment values to represent the cost center. For example, assign the balancing segment values of 001 and 013 with cost center PL04 to represent your manufacturing teams in the US and India. This creates 001-PL04 and 013-PL04 as the cost center reporting values. The cost center value of PL04 has a consistent meaning. This method requires fewer cost center values to be defined. However, it prevents construction of cost center hierarchies using trees where only cost center values are used to report results for a single legal entity. You must specify a balancing segment value in combination with the cost center values to report on a single legal entity.

Departments

A department is an organization with one or more operational objectives or responsibilities that exist independently of its manager. For example, although the manager may change, the objectives do not change. Departments have one or more workers assigned to them.

A manager of a department is typically responsible for:

- Controlling costs within their budget
- Tracking assets used by their department
- Managing employees, their assignments, and compensation

The manager of a sales department may also be responsible for meeting the revenue targets.

The financial performance of departments is generally tracked through one or more cost centers. In Oracle Fusion Applications, departments are defined and classified as Department organizations. Oracle Fusion Human Capital Management (HCM) assigns workers to departments, and tracks the headcount at the departmental level.

The granularity of cost centers and their relationship to departments varies across implementations. Cost center and department configuration may be unrelated, identical, or consist of many cost centers tracking the costs of one department.

FAQs for Manage Charts of Accounts

How can I use future accounting segments?

To plan for future growth in the business organization that requires additional segments in the chart of accounts. Extra segments can be added to the chart of accounts structure during your original implementation. All segments of the chart of accounts are required and have to be enabled. The unused segments can be assigned value sets that have a single value in the chart of accounts structure instance. The value is set as a default for that segment so that the extra segments are automatically populated when an account account combination is used.

Manage Value Sets

Value Sets: Explained

A value set is a group of valid values that you assign to a flexfield segment to control the values that are stored for business object attributes.



An end user enters a value for an attribute of a business object while using the application. The flexfield validates the value against the set of valid values that you configured as a value set and assigned to the segment.

For example, you can define a required format, such as a five digit number, or a list of valid values, such as green, red, and blue.

Flexfield segments are usually validated, and typically each segment in a given flexfield uses a different value set. You can assign a single value set to more than one segment, and you can share value sets among different flexfields.

Note: Ensure that changes to a shared value set are compatible with all flexfields segments using the value set.

The following aspects are important in understanding value sets:

- Managing value sets
- Validation
- Security
- Precision and scale
- Usage and deployment
- Protected value set data

Managing Value Sets

To access the Manage Value Sets page, use the Manage Value Sets task, or use the Manage Descriptive Flexfields and Manage Extensible Flexfields tasks for configuring a segment, including its value set. To access the Manage Values page, select the value set from the Manage Value Sets page, and click **Manage Values**. Alternatively, click **Manage Values** from the Edit Value Set page.

Validation

The following types of validation are available for value sets:

- Format only, where end users enter data rather than selecting values from a list
- Independent, a list of values consisting of valid values you specify
- Dependent, a list of values where a valid value derives from the independent value of another segment
- Subset, where the list of values is a subset of the values in an existing independent value set
- Table, where the values derive from a column in an application table and the list of values is limited by a WHERE clause

A segment that uses a format only value set doesn't present a list of valid values to users. Adding table validated value sets to the list of available value sets available for configuration is considered a custom task.

Note: For the Accounting Key Flexfield value sets, you must use independent validation only. If you use other validations, you can't use the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration.

Security

Value set security only works in conjunction with usage within flexfield segments. You can specify that data security be applied to the values in flexfield segments that use a value set. Based on the roles provisioned to users, data security policies determine which values of the flexfield segment end users can view or modify.



The application of value set security has the following conditions:

- At the value set level: The value set is the resource secured by data security policies. If a value set is secured, every usage of it in any flexfield is secured. It isn't possible to disable security for individual usages of the same value set.
- Applies to independent, dependent, or table-validated value sets.
- Applies mainly when data is being created or updated, and to key flexfield combinations tables for query purposes.
 Value set security doesn't determine which descriptive flexfield data is shown upon querying.
- Security conditions defined on value sets always use table aliases. When filters are used, table aliases are always
 used by default. When predicates are defined for data security conditions, make sure that the predicates also use
 table aliases.

For key flexfields, the attributes in the view object that correspond to the code combination ID (CCID), structure instance number (SIN), and data set number (DSN) cannot be transient. They must exist in the database table. For key flexfields, the SIN segment is the discriminator attribute, and the CCID segment is the common attribute.

Precision and Scale

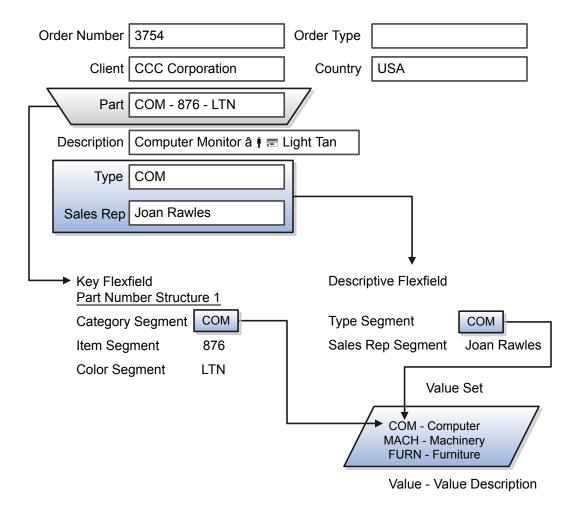
If the data type of a value set is Number, you can specify the precision (maximum number of digits user can enter) or scale (maximum number of digits following the decimal point).

Usage and Deployment

The usage of a value set is the flexfields where that value set is used. The deployment status of flexfields in which the value set is used indicates the deployment status of the value set instance.



The figure shows a value set used by a segment in a key flexfield and the context segment of a descriptive flexfield.



For most value sets, when you enter values into a flexfield segment, you can enter only values that already exist in the value set assigned to that segment.

Global and context-sensitive segment require a value set. You can assign a value set to a descriptive flexfield context segment. If you specify only context values, not value sets for contexts, the set of valid values is equal to the set of context values.

Protected Value Set Data

Application developers may mark some value sets as protected, indicating that you can't edit them.

You can edit only value sets that are not marked as protected. You can't edit or delete protected value sets. If the value set type supports values (such as independent, dependent or subset value sets), then you can't add, edit, or delete values.

Note: There is no restriction on references to protected value sets. Value sets, protected or not, may be assigned to any flexfield segment. Likewise, other value sets may reference protected value sets; for example, an unprotected dependent value set may reference a protected independent value set.



Related Topics

- Flexfields and Value Sets: How They Work Together
- Why can't I edit my flexfield or value set configuration?
- Defaulting and Deriving Segment Values: Explained
- What's the difference between a lookup type and a value set?

Chart of Accounts Values Sets: Critical Choices

A value set is the collection of account values that are associated with a segment of a chart of accounts structure instance. When creating values sets, consider the following critical choices:

- Module Designation
- Validation Type
- Format Assignments
- Security Rules
- Values Definition

Module Designation

The module designation is used to tag value sets in Oracle Fusion Applications and sets the value sets apart during upgrades and other processes. Chart of accounts value sets upgraded from Oracle E-Business Suite Release 12 generically bear the module value of **Oracle Fusion Middleware**. When creating value sets for a chart of accounts, the module can be specified as **Oracle Fusion General Ledger** to distinctly identify its intended use in an accounting flexfield, basically a chart of accounts.

Validation Type

Assign one of the following validation types to chart of accounts value sets:

- **Independent**: The values are independently selected when filling out the segment in the account combination.
- **Table Validated**: The values are stored in an external table to facilitate maintenance and sharing of the reference data.

▲ Caution: You must use Independent validation only for the Accounting Key Flexfield value sets. Other validations prevent you from using the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration. Dependent values sets are not supported.

Format Assignments

Value sets for chart of accounts must use the **Value Data Type** of **Character**. The **Value Subtype** is set to **Text**. These two setting support values that are both numbers and characters, which are typical in natural account segment values. Set the maximum length of the value set to correspond to the length of the chart of accounts segment to which it is assigned. Best practices recommend restricting values to **Upper Case Only** or **Numeric** values that are zero filled by default.

Security Rules

If flexfield data security rules are to be applied to the chart of accounts segment associated with the value set, the **Enable**Security check box must be checked for the assigned value set. In addition, assign a data security resource name to enable



creation of a data security object automatically for the value set. The data security object is used in the definition of flexfield data security rules.

Value Definition

Once these basic characteristic are defined for the value set, values can be added to the set in the Manage Values page.

- Set the values to conform to the value set length and type.
- Enter the value, its description, and its attributes including the **Enable** check box, **Start Date**, and **End Date**.
- Assign the following attributes: Parent or Summary check box, Posting is allowed, and Budgeting is allowed.

If the value set is used with a natural account segment, you must assign the **Natural Account Type**, with one of the following values: **Asset**, **Liability**, **Owner's Equity**, **Revenue**, or **Expense**.

▲ Caution: Assign account types carefully. If you assign an incorrect account type to a natural account segment value, accounting entries are recorded incorrectly and financial statements are inaccurate. Misclassified accounts are also potentially handled incorrectly at year end, with actual balances either getting zeroed out to retained earnings, or accumulating into the next year.

Other attributes used are **Third-Party Control Account**, **Reconciliation** indicator, and **Financial Category**, which is used with Oracle Transaction Business Intelligence reporting.

▼ Tip: Best practice is to define the values for the value set after the value set is assigned to a chart of accounts structure instance. Otherwise you aren't able to define the mandatory value attributes, such as the summary indicator, posting allowed, and account type for natural account segments. The attributes must be added after the value set is assigned to a chart of accounts structure instance.

Creating a Value Set for Your Chart of Accounts: Example

Create your value sets before creating your chart of accounts. A value set can be shared by different charts of accounts or across different segments of the same chart of accounts.

Scenario

You are creating a company value set to be used in your chart of accounts for your enterprise, InFusion America, Inc. Follow these steps:

- 1. Navigator > Setup and Maintenance > Manage Chart of Accounts Value Sets > Go to Task.
- 2. Click the **Create** icon on the toolbar of the Search Results table. The **Create Value Set** page opens.
- Enter a unique Value Set Code, InFusion America Company, and an optional Description, Company values for InFusion America Inc.
- 4. Select **General Ledger** from the list in the Module field.
- **5.** Select **Independent** as Validation Type.
 - Note: You must use Independent validation only for the Accounting Key Flexfield value sets. Other validations prevent you from using the full chart of accounts functionality, such as data security, reporting, and account hierarchy integration. Dependent values sets are not supported.
- 6. Select Character as the Validation Data Type.
- 7. Save and Close.



Manage Account Hierarchies

Trees: Overview

Trees are hierarchical data models that you can use to organize data, apply business rules, control data access, and improve performance while querying. For example, an application maintains data of an organization called Vision Corporation that has two departments: Marketing and Finance. The Finance department has two functional divisions: Receivables and Payables. You can define a tree for Vision Corporation to establish a hierarchy across its departments, and their respective functional divisions. You can use the hierarchy to manage data at various levels of the organization.

In the Setup and Maintenance work area, search for and use the Manage Trees and Tree Versions task to organize data into hierarchies.

Tree Structures

As the name suggests, tree structures provide you the framework to organize data such that you can establish a hierarchy for use by the tree. So, similar to a template, a tree structure guides the creation of a tree.

Tree

A tree is an instance of the tree structure. The root node is the topmost nodal point of a tree. Child nodes branch off from the root node. Child nodes at the same level, branching off from a common parent node, are called siblings. Leaves are details branching off from a node but not extending further down the tree hierarchy. You can create trees for multiple data sources and share them across applications.

Tree Versions

A tree by default has only one version. If required, you can create and maintain more than one editable tree version. At any point, only one tree version must be active. If you edit an existing version, it changes from active to draft. To use it again, you must set it to active. Similar to any other version control system, versions of trees are maintained to track all the changes that a tree undergoes in its life cycle.

Tree Labels

Tree labels are short names given to trees and tree structures. You can label the tree versions for better accessibility and information retrieval. When nodes are created in a tree, the existing tree labels are automatically assigned to the new tree nodes. You can use any table to store the labels and register the label data source with the tree structure.

In the Setup and Maintenance work area, search for and use the following tasks to work with trees:

- Manage Tree Structures: To create and update tree structures. You must first define a tree structure to create a tree.
- Manage Trees and Tree Versions: To create and update trees and their versions.
- Manage Tree Labels: To create and update tree labels.



Importing Segment Values and Hierarchies: Explained

Use the Import Segment Values and Hierarchies process to load segment values and hierarchies if you maintain chart of accounts reference data outside Oracle Fusion applications. You can load your segment values and hierarchies by populating two tables: GL_SEGMENT_VALUES_INTERFACE and GL_SEGMENT_HIER_INTERFACE, and running the Import Segment Values and Hierarchies process.

Note: You can load data to interface tables using predefined templates and the Load Interface File for Import scheduled process, which are both part of the External Data Integration Services for Oracle Cloud. For more information about file-based data import, see the File Based Data Import guide for your cloud services.

The GL_SEGMENT_VALUES_INTERFACE and GL_SEGMENT_HIER_INTERFACE tables

You can use GL_SEGMENT_VALUES_INTERFACE to load segment values and GL_SEGMENT_HIER_INTERFACE to load segment value hierarchies to Oracle Fusion applications. For more information about tables, see the Tables and Views for Oracle Financials Cloud guide.

Assigning Values for Columns in the GL_SEGMENT_VALUES_INTERFACE table

You must enter values in all columns of the interface table that require values for the Import Segment Values and Hierarchies process to be successful. The requirement includes all of the not null columns. Enter values in the following required columns of the interface table:

Column Name	Value
STATUS_CODE	Enter the value NEW to indicate that you are bringing new segment value data.
VALUE_SET_CODE	Enter the value set code for the segment values.
VALUE	Enter the segment value.
SUMMARY_FLAG	Select N if the segment value is a child value or Y if the segment value is a parent value.
ENABLED_FLAG:	Select Y to enable the segment value. Enter N to disable the segment value.
ACCOUNT_TYPE:	Enter the natural account type if the segment value is for natural account segment.
	A: Asset
	L: Liability
	E: Expense
	O: Owner's Equity
	R: Revenue
ALLOW_ POSTING_FLAG	Select Y if posting is allowed for this segment value. Select N if posting is not allowed.
OBJECT_ VERSION_ NUMBER	Enter default value of 1.



You can enter values for the following optional columns:

Column Name	Value
START_ DATE_ACTIVE	Enter the start date of the segment value
END_ DATE_ACTIVE	Enter the end date of the segment value.
THIRD_ PARTY_ CTRL_ACCOUNT	Enter the third-party control account value. Valid values are: CUSTOMER, SUPPLIER, R for Restrict Manual Journals, Y, and N.
FINANCIAL_ CATEGORY	Enter a financial category value for Oracle Transactional Business Intelligence reporting. Valid values are values defined in the FINANCIAL_ CATEGORY lookup type.
DESCRIPTION	Different description columns exist for different languages. To see segment value description in a different language installation, you populate the segment description for that language too.

The following columns should be left as null as Import Segment Values and Hierarchies process. The process uses them for internal processing or does not use them in the current release

- CREATION_DATE
- CREATED_BY
- LAST_UPDATE_DATE
- LAST_UPDATE_LOGIN
- LAST_UPDATED_BY
- SEGMENT_VALUE_INTERFACE_ID
- REQUEST_ID
- LOAD_REQUEST_ID

Assigning Values for Columns in the GL_SEGMENT_HEIR_INTERFACE table

You must enter values in all columns of the interface table that require values for the Import Segment Values and Hierarchies process to be successful. The requirement includes all of the not null columns. Enter values in the following required columns of the interface table:

Column Name	Value
STATUS_CODE	Enter the value NEW to indicate that you are bringing new hierarchy data.
VALUE_SET_CODE	Enter the value set code for the segment values.
TREE_CODE	Enter the hierarchy name (tree code).
TREE_VERSION NAME	Enter the hierarchy version name (tree version name).
TREE_ VERSION_ START_ DATE_ACTIVE	Enter the date that the tree version is activated.



Column Name	Value
TREE_ VERSION_ END_ DATE_ACTIVE	Enter the date that the tree version is inactivated.
VALUE	Enter the segment value.
PARENT_VALUE	Select N if the segment value is a child value or Y if the segment value is a parent value.
DEPTH	Enter the depth of the hierarchy which shows the many ancestors the segment value has in the hierarchy.
OBJECT_VERSION_NUMBER	Enter default value of 1.

The following columns should be left as null as Import Segment Values and Hierarchies process. The process uses them for internal processing or does not use them in the current release.

- CREATION DATE
- CREATED BY
- LAST_UPDATE_DATE
- LAST UPDATE LOGIN
- LAST_UPDATED_BY
- SEGMENT_VALUE_INTERFACE_ID
- REQUEST_ID
- LOAD REQUEST ID

Related Topics

- External Data Integration Services for Oracle Cloud: Overview
- Loading Data to the Segment Value and Hierarchies Interface Tables: Explained
- Tables and Views for Oracle Financials Cloud
- File Based Data Import for Oracle Financials Cloud

Publishing an Account Hierarchy: Example

Publish an account hierarchy to finalize a new or edited account hierarchy. Account hierarchies organize data and enable the creation of groups and rollups of information that exist within an organization. After creating or editing an accounting hierarchy, run the Publish Account Hierarchies, Maintain GL Value Sets, and Maintain Chart of Account Hierarchies processes before using the new or changed account hierarchy.

Scenario

You have made changes to your InFusion America account hierarchy. Follow these steps to publish your account hierarchy.

1. Navigator > Setup and Maintenance.



- 2. Enter the Publish Account Hierarchies into the Name field. Click Search.
- 3. Click the Go to Task to open the Publish Account Hierarchies page.
- 4. Search for your InFusion America account hierarchy.
- **5. Expand** to open the versions of your hierarchy.
- 6. Click the **Publish** button on the tool bar.

Note: For a new hierarchy, scroll over to verify the Publish check box is checked.

Hierarchies for Cross Validations, Revaluations, and Chart of Accounts Mapping: Explained

Hierarchies you create for reporting or allocations may not be suitable for setting up cross validations, revaluation, and chart of accounts mapping rules.

For example, if you must enforce a cross validation rule that a set of 20 departments is applicable only to a certain company, then you can group the 20 departments under a hierarchy node and refer to that hierarchy node in the cross validation rule.

Cross Validations, Revaluations, and Chart of Accounts Mapping

For hierarchies for cross validations, revaluations, and chart of accounts mapping:

- Create hierarchies for cross validations, revaluation, or chart of accounts mapping within one hierarchy. Associate
 this hierarchy to the chart of accounts segment instance in the chart of accounts instance setup page. You can only
 associate one hierarchy with the chart of accounts instance. You can create separate root nodes for each segment
 of the hierarchy.
- Create an account hierarchy for each of the segments that are used in the rule set ups. Depending on your requirements, not all segments may need a hierarchy.
- Use the same child to roll up to different parents if you need different roll ups for cross validations, revaluations, and chart of accounts mapping.
- Flatten and audit the hierarchy after the hierarchies are complete and set it to an active status.
- Associate the hierarchy to the chart of accounts segment instance.
- Redeploy the accounting flexfield after the chart of accounts instance is updated.

If you have the duplicate segment values in a hierarchy, you should not publish the hierarchies to Oracle Essbase.

Naming Financial Reporting and Allocations Hierarchies: Points to Consider

You must publish hierarchies for financial reporting and allocations to Oracle Essbase cubes. When a hierarchy is published to an Essbase cube, a fully qualified member name is generated for each node in the hierarchy. This fully qualified member name is used to refer to the node in the hierarchy in financial reports and allocation rules.

Each combination of tree plus tree version is published as a different account hierarchy to Essbase cubes.

When a chart of accounts child or parent value is assigned to multiple hierarchy versions, the fully qualified member name includes the hierarchy version name.



Member Names and Financial Reporting and Allocations Hierarchies

You must consider the implications to your financial reports, allocation rules, and Smart View templates if the member name changes due to hierarchy changes. The points to considered are:

- The member name for a member can change if you originally published a single hierarchy to a cube and later published another hierarchy. The change happens if the member includes the same chart of accounts value.
- If the fully qualified member name changes, you must update existing financial reports, allocation rules, and Smart View templates that refer to that member. Otherwise, such processes have errors.

For example, only a single hierarchy version, V1, existed when you defined a financial report. The hierarchy version includes the chart of accounts value of 500 for a cost center. The path does not include a hierarchy version name, since there is only one hierarchy version in the cube. The fully qualified name path is 500.

Later, you publish a new hierarchy version, V2. The fully qualified member name changes since there are now two versions and cost center 500 is associated with two hierarchy versions, V1 and V2. When selecting the value 500, the cube has logic to uniquely identify the member. The fully qualified member name for the two hierarchy versions is now:

- [All VF Cost Centers-**V1**].[999].[500]
- [All VF Cost Centers-**V2**].[999].[500]
- Note: The fully qualified member can also be shortened by the cube logic, such as [Cost_Center]@[Cost Center Level 29 Code]|[500].

Update configurations, such as financial reports, allocation rules, or Smart View queries that referenced the original name path of 500 with the correct name.

Related Topics

• Initial Publishing of Account Hierarchies: Points to Consider

Maintain Account Hierarchies: Overview

This topic describes the best practices of maintaining and publishing the account hierarchies. Account hierarchies are used throughout Oracle Fusion General Ledger for creating financial reports, Smart View inquiries, allocation definitions, cross validation rules, and revaluation definitions.

Related Topics

· Account Hierarchy Trees: Explained

Maintenance of Hierarchies: Explained

As your organization changes or there are updates to your chart of accounts values, you can make updates to your hierarchy versions. Changes must be in the two hierarchy versions to minimize errors related to member name paths in financial reports, allocation rules, or Smart View queries. Both hierarchy versions must be published to the balances cube.

To keep an audit history, copy the hierarchy version and use the copy as the version for audit history. After you copy the hierarchy version, modify that same version to use as the latest version. You do not have to change reports.



Hierarchy Versions Configuration

Perform the following steps before updating the current hierarchy version:

- 1. Copy the current hierarchy version to create cost center 2011 to keep an audit history for year 2011.
- 2. Update the effective date for the **Current** hierarchy version to the current period. Set the effective dates for the copied hierarchy version to what the past dates were.
- 3. Make all of your current hierarchy changes to the **Current** hierarchy version.
- **4.** Set the status to **Active** again.
- 5. Delete the Baseline hierarchy version in the **Manage Account Hierarchies** page. You do not unpublish the hierarchy from the cube, as long as you follow the naming conventions discussed in the following steps.
- **6.** Copy the **Current** hierarchy version and name it **Baseline**. You must name it the same exact name because Baseline still exists in the cube. The procedures overwrite the hierarchy version named **Baseline** in the cube.
- 7. Use effective dates that do not overlap and that are before periods used by previously active hierarchy versions.
- 8. Set the status for the Baseline hierarchy version to **Active**.
- 9. Publish both hierarchy versions, **Current** and **Baseline**, to their respective cube.
- **10.** Publish the cost center 2011 to the cube if the version is still needed for financial reporting, Smart View queries, or allocation rules.
- 11. Review the published results using Smart View.

You have the following three versions:

- 1. The new copied version: This version is used to track the history of the old master version, for example, cost center 2011 and have been end dated.
- 2. Current version: This version has been modified to reflect the new hierarchy and is now active.
- 3. Baseline version: This version should continue to be the duplicate of the master version. Rather than modifying this manually, delete the old baseline version and create another copy of the master. Modify the master version. These steps should be followed each time when there are hierarchy changes that must be published.
- Note: Best practices recommend that if there are changes, you update the hierarchies before completing period close and financial reporting.

Maintaining Hierarchies: Examples

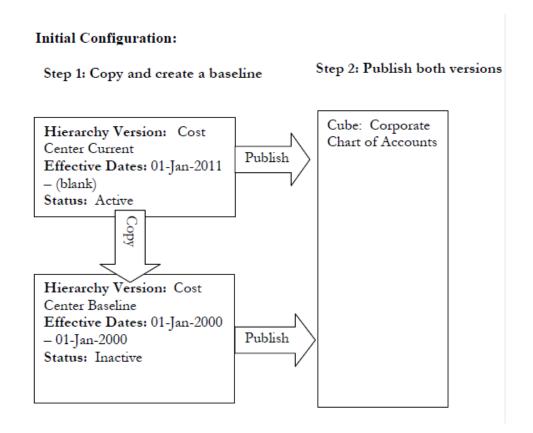
You can update the hierarchy for any changes that must be reflected in 2012. The following example illustrates how to maintain your hierarchy.

Your organization, Vision Operations:

- Decided in 2010 to begin using Oracle Fusion General Ledger, effective January 1, 2011.
- Uses a single chart of accounts named **Corporate Chart of Accounts**. Since you only have one chart of accounts, this is also the name of your cube.
- Uses hierarchy to capture its cost center roll ups by line of business. The name of this hierarchy is named Cost Centers Hierarchy.



The following graphic displays the relationships between the hierarchy versions and the cube:



Maintaining and Updating Your Hierarchy

At the end of 2011, your organization, Vision Operations, makes organizational changes to its lines of business. You add new cost centers and a new line of business. As a result, you must update your cost center hierarchies to ensure that financial reporting reflects the new organization hierarchy.

Do not make changes to the Cost Center Master hierarchy version if no account value changes have been made and the current hierarchies are working.

However, since there are changes you must:

- 1. Copy and backup the **Cost Center Current** hierarchy version to maintain history.
- 2. Make changes to the **Cost Center Current** hierarchy version and change it to the new effective dates.
- **3.** Delete the hierarchy version named **Baseline**. You do not unpublish the hierarchy from the cube, as long as you follow the naming conventions discussed in the following steps.
- 4. Copy the **Current** hierarchy version after changes are completed and name it **Baseline**.
- **5.** Publish the **Current** hierarchy version again.
- **6.** Publish the new **Baseline** hierarchy version again.

After publishing both versions, there are still only two hierarchy versions in the cubes. The version name must always be the same across all periods. The name is reference in the financial reports. allocation rules, and Smart View queries. For example, Cost Center Current and Cost Center Baseline. Do not have a version named 2011 Cost Center, or 2012 Cost Center Current, or 2012 Cost Center Baseline.

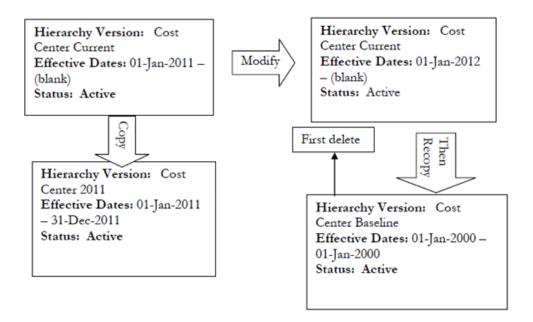


This example assumes an annual change. It can also be quarterly, monthly, or as needed. Follow these steps for any hierarchy changes at any time for even regular maintenance, such as adding values or moving values in the hierarchy. Best practices recommend keeping your current and baseline hierarchy versions in sync.

The following graphic displays steps to update your hierarchy versions:

Step 1: (Optional) Copy Cost Center Current to Cost Center 2011 to maintain history

Step 2: Modify Cost Center Current to create new version, Delete old Cost Center Baseline Create a new Cost Center Baseline



FAQs for Manage Account Hierarchies

How can I manage and review account hierarchies?

View segment value descriptions when creating, editing, or reviewing account hierarchies online. You can also export account hierarchies to a spreadsheet to review, analyze, and report offline.

To export all nodes in the hierarchies to a spreadsheet, expand all nodes first from **View -> Expand All** in the **Manage Account Hierarchies: Specify Nodes** page.

When do I perform row and column flattening for tree versions manually?

Row Flattening and Column Flattening for tree versions need to be manually submitted under the following scenarios.

When a new tree or tree version is defined.



- If the tree or tree version has any change made to it that would alter the flattened hierarchy data. For example, adding, moving, and duplicating members. Basically, anything that impacts the flattened hierarchy data.
- If Data Relationship Management (DRM) is used with Oracle Fusion General Ledger, after running the Load Account Values and Hierarchies process from the Scheduled Process work area to load chart of accounts reference data from DRM to Oracle Fusion General Ledger.
- When range based hierarchy assignments are supported, if new values are added that are within the account range assigned to the hierarchy.
 - Note: In this case, try submitting the flattening programs using the Online Mode first. If the newly inserted child account value does not appear to be included in the flattened hierarchy data, then the flattening program has to be submitted using the Force Flattening Mode, instead of Online Flattening. An issue with the incremental flattening programs can prevent them from picking up this type of hierarchy change, so full flattening can be required.
- Note: Column flattening processed data is primarily relevant to Oracle Fusion Transactional Business Intelligence, but there should not be any adverse impact to run both row and column flattening processes.

Manage Tree Structures

Tree Structures: Explained

A tree structure defines the hierarchy for creating trees and prescribes rules based on which trees are created, versioned, and accessed. You can associate multiple data sources with a tree structure. A tree is an instance of this hierarchy. Every tree structure can contain one or more trees.

You can create tree structures specific to an application but you can share tree structures across applications. If you apply version control to the tree structure, it is carried over to the trees that are based on the tree structure. Each tree version contains at least one root node. Occasionally, a tree version may have more than one root node.

An administrator controls the access to tree structures through a set of rules that are periodically audited for validity.

Tree Structure Definition: Points to Consider

While creating a tree structure, you must specify important details on the Create Tree Structure: Specify Definition page. As the source of the tree structure, you may either select the predefined tree structures and proceed with the definition or create custom tree structures.

Tree Node Selection

The data in **Tree Node** table maps to the data in nodes of the tree structure. You must select the correct and most appropriate tree node table to define the tree structure, based on which you establish the tree hierarchy. This selection also affects the level of security that is set on a tree node and its child entities.

Tree Sharing Mode

Use the following options to determine the mode of sharing a tree structure across the applications.

- Open: Indicates that the tree is associated with all reference data sets.
- Set ID: Indicates that the tree is associated with a specific reference data set.



Customization

You can customize the predefined tree structures as well as those you create. However, customizing a predefined tree structure is restricted and permitted through additional privileges. Customization is limited to specific tree nodes and below in the tree hierarchy.

Multiple Tree Versions

Although multiple tree versions can exist together, Oracle recommends only one version be active at any given time. However, if required, you can have more tree versions to be in the active state for the same date range. You can use this flexibility to select the tree version you want to implement.

Managing Tree Structures: Points to Consider

You can create, edit, and delete tree structures. You can also change the status of a tree structure and audit the changes.

Creating and Editing Tree Structures

When you edit an active tree structure, the status of the tree structure and all associated trees and their versions changes to draft. To reuse a tree structure, create a copy of the tree without copying the associated trees and tree versions. After making changes, set the status again to active. If you delete a tree structure, all the associated trees and tree versions are automatically deleted.

For information about working with the offering-specific predefined tree structures, refer to the relevant product documentation.

Status

When you change the status of a tree structure, the status of the trees and tree versions associated with that tree structure also changes.

The following table lists the different statuses of a tree structure.

Status	Meaning
Draft	In a modified state, or not yet in use.
Active	In use, indicating that one or more trees or tree versions are created from the tree structure.
Inactive	Not in use.

Tree Structure Audit Results: Explained

Use the tree structure audit results to verify the tree structure's correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- The result of the validation, including a detailed message
- Corrective actions to take if there are any validation errors

Running an Audit

Setting the status of a tree structure to active automatically triggers an audit of that tree structure. To manually trigger an audit, select Audit from the Actions menu on the Manage Tree Structures page. The Tree Structure Audit Result table shows a list of validations that ran against the selected tree structure.



Audit Validators

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

Validator	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
Restrict By Set ID	Manage Tree Structures: Specify Data Sources	If you select the Reference Data Set check box for the Restrict Tree Node List of Values Based on option, each of its data source view objects must have a reference data set attribute.	Even when the check box is selected, one or more data source view objects doesn't contain a reference data set attribute.	If reference data set restriction is required for this tree structure, include a reference data set attribute on all data sources. Otherwise, deselect the check box.
Available Label Data Sources	Manage Tree Structures: Specify Data Sources	If you select a list item from Labeling Scheme to specify a labeling scheme, the label data source view object specified for each data source must be accessible. Also, the primary keys must be valid. This restriction doesn't apply if you select None from the list.	 Any of the specified label data source view objects doesn't exist. Any of the specified label data source view objects doesn't have primary keys. When a label data source view object is initially defined, the database registers the primary keys for the view object. If the view object is later modified such that its primary keys no longer match the primary keys that were registered earlier, this validation fails. 	Correct the specified label data source view object. Correct the primary keys of the specified label data source view object. Do one of the following: Correct the primary keys in the label data source view object to match the primary keys that were earlier registered in FND_TS_DATA_SOURCE To match the primary keys registered in that table to match the new view object definition.
Row Flattened Table Name	Manage Tree Structures: Specify Performance Options	You must specify a valid row flattened table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE_RF or a custom table.	 The specified table doesn't exist in the database. The specified table doesn't contain the same columns as the 	Correct the row flattened table definition.



Validator	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
			FND_TREE_NODE_R I table.	Ŧ
Available Data Sources	Add Data Source	Each data source view object specified for the tree structure must be accessible, and all its primary key attributes must be valid.	 Any of the specified data source view objects doesn't exist. When you define a data source view object, keep the Use non-defined primary key columns check box deselected. The database automatically registers the primary keys for the view object. Select this check box if you want the database to register the primary keys you specify. However, if the registered primary keys contain any duplicates, this validation fails. The Use non-defined primary key columns check box is selected in a data source, but the list of specified primary key columns doesn't match the primary keys defined in the corresponding data source view object. Any common attribute that exists in both the data source view object and the tree node view object isn't of the same data type in both view objects. 	Correct the specified data source view object. Correct the duplicate column in the registered primary keys. Correct the primary keys of the specified data source view object. Correct any mismatch in data types.
Column Flattened Table Name	Manage Tree Structures: Specify Performance Options	You must specify a valid column flattened table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE_CF or a custom table.	 The specified table doesn't exist in the database. The specified table doesn't contain the same columns as the 	Correct the column flattened table definition.



Validator	Page	Description (what is validated)	Possible Cause for Validation Failure	Suggested Corrective Action
		FND_TREE_NODE_CF table.		
Restrict by Date	Manage Tree Structures: Specify Data Sources	If you select the Date Range check box for the Restrict Tree Node List of Values Based on option for a tree structure, each of its data source view objects must have effective start date and end date attributes. This validation doesn't take place when the check box isn't selected.	Even when the check box is selected, one or more of its data source view objects doesn't contain effective start date and end date attributes.	If the date restriction is required for this tree structure, include the effective start date and effective end date attributes on all data sources. Otherwise, deselect the check box.
Tree Node Table Name	Manage Tree Structures: Specify Definition	You must specify a valid tree node table for the tree structure. It can either be the standard row flattened table FND_TREE_NODE or a custom table.	No table is specified in the Tree Node Table field. The specified table doesn't exist in the database. The specified table doesn't contain the same columns as the FND_TREE_NODE table.	Correct the tree node table definition.

Specifying Data Sources for Tree Structures: Points to Consider

The data sources provide the items for establishing hierarchy in a tree structure. In the tree management infrastructure, these data sources are Oracle ADF business components view objects, which are defined by application development.

Labeling Schemes

Selecting a labeling scheme determines how the tree nodes are labeled. You may select a labeling scheme to assign at the data source level, at the parent node level, or keep it open for customers assignment. You may also choose not to have any labeling scheme. However, if you decide to use any of the labeling schemes, select the following additional options, to restrict the list of values that appear under the selected tree node.

- Allow Ragged Nodes: To include nodes that have no child nodes, and are shorter than the remaining nodes in the
 entire hierarchy.
- Allow Skip Level Nodes: To include nodes that are at the same level but have parent nodes at different levels.

Restriction of Tree Node Values

You can decide the depth of the tree structure by selecting an appropriate value from the list. Keeping the depth limit open renders an infinite list of values.

Using the following options, you can restrict the list of values that appear for selection under a specific tree node.

- **Date Range**: Specifies whether a selection of nodes should be restricted to the same date range as the tree version.
- Allow Multiple Root Nodes: Allows you to add multiple root nodes when creating a tree version.



• Reference Data Set: Specifies whether a selection of nodes should be restricted to the same set as the tree.

Data Source Values and Parameters

Tree data sources have optional data source parameters with defined view criteria and associated bind variables. You can specify view criteria as a data source parameter when creating a tree structure, and edit the parameters when creating a tree. Multiple data sources can be associated with a tree structure and can have well-defined relationships among them.

Note: Parameter values customized at the tree level override the default values specified at the tree-structure level.

The data source parameters are applied to any tree version belonging to that data source, when performing node operations on the tree nodes. Data source parameters also provide an additional level of filtering for different tree structures. The tree structure definition supports three data source parameter types.

- Bound Value: Captures any fixed value, which is used as part of the view criteria condition.
- Variable: Captures and binds a dynamic value that is being used by the data source view object. This value is used by the WHERE condition of the data flow.
- View Criteria: Captures the view criteria name, which is applied to the data source view object.

You can also specify which of the data source parameters are mandatory while creating or editing the tree structure.

View objects from the Oracle ADF business components are used as data sources. To associate the view object with the tree structure, you can pick the code from Oracle ADF business component view objects and provide the fully qualified name of the view object, for example, oracle.apps.fnd.applcore.trees.model.view.FndLabelVO.

Specifying Performance Options for a Tree Structure: Points to Consider

Tree structures are heavily loaded with data. As a tree management guideline, use the following settings to improve performance of data rendering and retrieval.

- Row Flattening
- Column Flattening
- Column Flattened Entity Objects
- Bl View Objects

Row Flattening

Row flattening optimizes parent-child information for run-time performance by storing additional rows in a table for instantly finding all descendants of a parent without initiating a CONNECT BY query. Row flattening eliminates recursive queries, which allows operations to perform across an entire subtree more efficiently.

To store row flattened data for the specific tree structure, users can either use the central <code>fnd_tree_node_rf</code> table or they can register their own row flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a row-flattened table contains an additional row with Corporation directly being the parent of Region (Corporation-Region).

Column Flattening

Column flattening optimizes parent-child information for runtime performance by storing an additional column in a table for all parents of a child.

To store column flattened data for the specific tree structure, users can either use the central FND_TREE_NODE_CF table or they can register their own column flattened table. For example, in a table, if Corporation is the parent of Sales Division (Corporation-Sales Division), and Sales Division is the parent of Region (Sales Division-Region), a flattened table in addition



to these columns, contains three new columns: Region, Sales Division, and Corporation. Although positioned next to each other, the column Region functions at the lower level and Corporation at the higher level, retaining the data hierarchy.

Column Flattened Entity Object

In the absence of a column-flattened table, if you need to generate the business component view objects for your tree structure for the flattened table, use the tree management infrastructure to correctly provide the fully qualified name of the entity object for the column flattened table.

BI View Object

View objects from Business Intelligence can be used as data sources, eliminating the need to create new types of data sources. This field is to store the fully qualified name for the BI view object generated by the tree management for business intelligence reporting and usage The BI view object is a combination of the tree data source and column flattened entity. Using this option prevents data redundancy and promotes greater reuse of existing data, thereby improving the performance of the tree structure.

Search View Object

Specify the full name of the view object for the tree node to ensure that search operations performed on the tree node are efficient.

Manage Tree Labels

Tree Labels: Explained

Tree labels are tags that are stored on tree nodes. You can store labels in any table and register the label data source with the tree structure. When a labeling scheme is used for trees, the selected labels are stored in the tree label entity, and each tree node contains a reference to a tree label in the labeling scheme.

The following table lists the three ways in which tree labels are assigned to the tree nodes.

Labeling Scheme	Description
Level	Labels that are automatically assigned based on the data source to which the tree node belongs. A level label points to a specific data source. For example, in a tree that reflects the organizational hierarchy of an enterprise, all division nodes appear on one level and all department nodes on another.
Group	Labels that you can arbitrarily assign to tree nodes.
Depth	Labels that are automatically assigned based on the depth of the tree node within the tree. No manual assignment is performed.
	Note: In an unbalanced hierarchy, a level may not be equal to depth.



Manage Trees and Tree Versions

Managing Trees and Tree Versions: Points to Consider

You can create and edit trees and tree versions depending upon the requirement. A tree can have one or more tree versions. When changes are made to an existing tree, a new version is created and published.

Creating and Editing Trees

Trees are created based on the structure defined in the tree structure. You can create trees, modify existing trees, and delete trees. If you want to copy an existing tree, you can duplicate it. You can also select and copy the associated tree versions.

Creating a tree involves specifying the tree definition and specifying the labels that are used on its nodes. If the selected tree structure has data sources and parameters defined for it, they appear on the page allowing you to edit the parameter values at the tree node level.

Note: Parameter values customized at the tree level will override the default values specified at the treestructure level.

Creating and Editing Tree Versions

Tree versions are created at the time of creating trees. Each tree must contain a version.

Editing an existing tree provides you with the option of updating the existing version. You can also edit the existing version that lies nested under the tree in the search results.

When you edit a tree version bearing Active status, the status changes to Draft until the modifications are saved or canceled.

Tree Version Audit Results: Explained

Use the tree version audit results to verify the tree version's correctness and data integrity. The audit results include the following details:

- The name of the validator, which is a specific validation check
- · The result of the validation, including a detailed message
- · Corrective actions to take if there are any validation errors

Running an Audit

An audit automatically runs whenever a tree version is set to active. You can also manually trigger an audit on the Manage Trees and Tree Versions page, using **Actions - Audit**. The Tree Version Audit Result table shows a list of validations that ran against the selected tree version.

Validation Details

The following table lists the validators used in the audit process and describes what each validator checks for. It also lists possible causes for validation errors and suggests corrective actions.

Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
Effective Date	The effective start and end dates of the tree version must be valid.	The effective end date is set to a value that is not greater than the effective start date.	Modify the effective start and end dates such that the effective



Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
			start date is earlier than the effective end date.
Root Node	On the Manage Tree Structures: Specify Data Sources page, if the Allow Multiple Root Nodes check box for the Restrict Tree Node List of Values Based on option is not selected, and if the tree structure is not empty, the tree version must contain exactly one root node. This validation does not take place if the check box is selected.	Even if the check box is deselected, the tree version has multiple root nodes.	Modify the tree version such that there is exactly one root node.
Data Source Max Depth	For each data source in the tree structure, on the Data Source dialog box, if the data source is depth-limited, the data in the tree version must adhere to the specified depth limit. This validation doesn't apply to data sources for which the Maximum Depth field is set to Unlimited .	The tree version has data at a depth greater than the specified depth limit on one or more data sources.	Modify the tree version such that all nodes are at a depth that complies with the data source depth limit.
Duplicate Node	On the Data Source dialog box, if the Allow Duplicates check box isn't selected, the tree version must not contain more than one node with the same primary key from the data source. If the check box is selected, duplicate nodes are permitted.	Even when the check box is deselected, the tree version contains duplicate nodes.	Remove any duplicate nodes from the tree version.
Available Node	All nodes in the tree version must be valid and available in the underlying data source.	 A node in the tree version doesn't exist in the data source. Deleting data items from the data source without removing the corresponding nodes from the tree version can result in orphaned nodes in the tree version. For example, if you added node A into your tree version, and subsequently deleted node A from the data source without removing it from the tree version, the validation fails. The tree version contains a tree reference node, which references another tree version that does not exist. 	Remove any orphaned nodes from the tree version. Update tree reference nodes so that they reference existing tree versions.



Validator	Description (what is checked)	Possible Cause for Validation Failure	Suggested Corrective Action
Node Relationship	All nodes must adhere to the relationships mandated by the data sources registered in the tree structure.	The tree structure has data sources arranged in a parent-child relationship, but the nodes in the tree don't adhere to the same parent-child relationship. For example, if the tree structure has a Project data source with a Task data source as its child, Task nodes must always be under Project nodes in the tree version. This validation fails if there are instances where a Project node is added as the child of a Task node.	Modify the tree version such that the nodes adhere to the same parent-child relationships as the data sources.
SetID Restricted Node	On the Manage Tree Structures: Specify Data sources page, if the Set ID check box is selected to enable the Restrict Tree Node List of Values Based on option for each tree node, the underlying node in the data source must belong to the same reference data set as the tree itself. This restriction doesn't apply when the check box is not selected.	Even when the check box is selected, the tree version has nodes whose data source values belong to a different reference data set than the tree.	Modify the tree version such that all nodes in the tree have data sources with reference data set matching that of the tree.
Label Enabled Node	On the Manage Tree Structures: Specify Data Sources page, if a labeling scheme is specified for the tree structure by selecting a list item from the Labeling Scheme list box, all nodes must have labels. This restriction doesn't apply when you select None from the Labeling Scheme list box.	The tree structure has a labeling scheme but the tree version has nodes without labels.	Assign a label to any node that doesn't have a label.
Date Restricted Node	On the Manage Tree Structures: Specify Data Sources page, if the Date Range check box is selected to enable the Restrict Tree Node List of Values Based on option for a tree structure, each node in the underlying data source must have an effective date range same as the effective date range of the tree version. This restriction doesn't apply if the check box isn't selected.	Even when the check box is selected, there are data source nodes that have a date range beyond the tree version's effective date range. For example, if the tree version is effective from Jan-01-2012 to Dec-31-2012, all nodes in the tree version must be effective from Jan-01-2012 to Dec-31-2012 at a minimum. It is acceptable for the nodes to be effective for a date range that extends partly beyond the tree version's effective date range (for example, the node data source value is effective from Dec-01-2011 to Mar-31-2013). It isn't acceptable if the nodes are effective for none or only a part of the tree version's effective	Ensure that all nodes in the tree version have effective date range for the effective date range for the tree version.



Validator	Description (what is checked)	Possible Cause for Validation Failure date range (for example, the node data source value are effective only from Jan-01-2012 to June-30-2012).	Suggested Corrective Action
Multiple Active Tree Version	On the Manage Tree Structures: Specify Definition page, if the Allow Multiple Active Tree Versions check box isn't selected for the tree structure, there must not be more than one active tree version under a tree at any time. This restriction doesn't apply if the check box is selected.	Even when the check box isn't selected, there is more than one active tree version in the tree for the same date range.	Set no more than one tree version to Active within the same date range and set the others to inactive or draft status.
Range Based Node	On the Data Source dialog box, if the Allow Range Children check box isn't selected, range-based nodes are not permitted from that data source. This restriction doesn't apply if the check box is selected.	Even when the check box isn't selected, there are range-based nodes from a data source.	Ensure that any range nodes in your tree version are from a data source that allows range children.
Terminal Node	On the Data Source dialog box, if the Allow Use as Leaves check box isn't selected, values from that data source can't be added as leaves (terminal nodes) to the tree version. This restriction doesn't apply if the check box is selected.	Even when the check box isn't selected, values from a data source are added as leaf nodes (terminal nodes).	Modify the tree version such that all terminal nodes are from data sources for which this check box is selected.
Usage Limit	On the Data Source dialog box, if the Use All Values option is selected to set the Usage Limit for the data source, every value in the data source must appear as a node in the tree. This restriction doesn't apply if None option is selected.	Even if the Use All Values option is selected, there are values in the data source that aren't in the tree version.	For each data source value that isn't yet available, add nodes to the tree version.

Trees and Data Sources: How They Work Together

Data sources are the foundation of tree management. Tree structures, trees, and tree versions establish direct and real-time connectivity with the data sources. Changes to the data sources immediately reflect on the **Manage Trees and Tree Versions** page and wherever the trees are being used.

Metadata and Data Storage

Tree structures contain the metadata of the actual data and the core business rules that manifest in trees and tree versions. You can select and enable a subset of trees to fulfill a specific purpose in that application.



Access Control

Source data is mapped to tree nodes at different levels in the database. Therefore, the changes you make to the tree nodes affect the source data. Access control set on trees prevents unwanted data modifications in the database. Access control can be applied to the tree nodes or anywhere in the tree hierarchy.

Adding Tree Nodes: Points to Consider

Tree nodes are points of data convergence where a tree branches into levels. Nodes are the building blocks of a tree structure and are attached to tree versions. Whenever you create or edit a tree version, you need to specify its tree node.

In the Setup and Maintenance work area, search for the Define Trees task and access the Manage Trees and Tree Versions page.

Managing Tree Nodes

You can create, modify, or delete tree nodes on the **Tree Version: Specify Nodes** page. To add a tree node, ensure that the tree structure with which the tree version is associated is mapped to a valid data source. You can also duplicate a tree node if the multiple root node feature is enabled.

Node Levels

Usually, the nodes at a particular level represent similar information. For example, in a tree that reflects the organizational hierarchy, all nodes representing divisions appear at one level and all the department nodes on another. Similarly, in a tree that organizes a user's product catalog, the nodes representing individual products might appear at one level and the nodes representing product lines on the immediate higher level.

The following node levels are in use:

- Root node: The topmost node in the tree structure
- Parent node: The node that branches off into other nodes
- Child node: The node that is connected to a node higher in hierarchy (parent node)
- Sibling node: Nodes that are at the same level and belong to the same parent node
- Leaf node: Entities branching off from a node but not extending further down the tree hierarchy

Node Types

A tree node has the following node types.

- Single: Indicates that the node is a value by itself.
- Range: Indicates that the node represents a range of values and possibly could have many children. For example, a tree node representing account numbers 10000 to 99999.
- Referenced Tree: Indicates that the tree node is actually another version for the tree based on the same tree structure, which is not physically stored in the same tree. For example, a geographic hierarchy for the United States can be referenced in a World geographic hierarchy.

Manage Accounting Combinations



General Ledger Misclassified Accounts Validations Diagnostic Test

The Oracle Fusion General Ledger Misclassified Accounts Validations diagnostic test identifies the code combinations that have a different segment label than the account segment used in the code combination. For example, the account segment is classified as revenue, but the code combination contain that account segment is classified as expense. The test also identifies if there are balances in the previous year end and the entered currency for those balances.

One sign that an account is misclassified is when opening a new fiscal year, the retained earnings account does not reflect the correct balances. Another warning sign is when balance sheet account balances go to zero after opening the new fiscal year, instead of carrying forward the previous year's balances.

Segment labels identify certain segments in your chart of accounts structure and assign special functionality to those segments. The Natural Account segment label classifies a segment value with a specific account type. The available account types are: asset, liability, expense, revenue, or owner' equity. The default account type is expense. This classification is used in General Ledger to perform account transfers during the year end closing process.

The natural account segment is mapped to the Financial Category dimension in the General Ledger balances cubes to enable reporting and transactional queries in the General Accounting and other dashboards. This functionality uses Oracle Transaction Business Intelligence (OTBI) to analyze and drill into expense and revenue transactions. The natural account segment label is required. Misclassification can have serious consequences, requiring solutions from Oracle Fusion iSupport to correct errors if the segment value has been used to record accounting transactions.

Run Misclassified Accounts Validation diagnostic test when experiencing issues with General Ledger code combinations, such as the balance of balance sheet accounts not rolling forward into the new year or unexpected balances in the retrained earnings account. Review the diagnostic output and follow the suggested actions. It is recommended that you run this test periodically from the Enterprise Scheduler.

Required Parameters

Provide Ledger ID

Select a ledger ID from the list of values to restrict the test by ledger.

Provide Period Name

Select an accounting period from the list of values to restrict the test to one period.

Note: Use the first period of the accounting calendar to verify accounts are classified correctly before processing transactions.

Results

The test results include error and warning messages, followed by suggested actions, identify possible problem areas. The test output shows all misclassified accounts for the specified ledger and accounting period. The test output data is categorized by the following criteria: ledger, chart of accounts, accounting calendar, journals, average balance budgetary control, and misclassified accounts references. The account details used to identify the misclassified accounts are segment name, segment number, natural account segment, and value set.

Note: Changing the account type of the segment value in the flexfield values page does not change it in the account code combination page. Make all changes in both pages, if necessary.



Maintain Segment Value Attributes

Segment Labels: Explained

Segment labels identify certain segments in your chart of accounts structure and assign special functionality to those segments. Segment labels were referred to as flexfield qualifiers in Oracle E-Business Suite (EBS). Best practice is to assign each segment label one time within the chart of accounts structure. Here are the segment labels that are available to use with the chart of accounts structure.

Caution: Validations are not performed when segment labels are assigned, so verify that all are assigned correctly before using your chart of accounts.

Balancing

Ensures that all journals balance for each balancing segment value or combination of multiple balancing segment values. You can secure access to your primary balancing segment values only with data access sets. The General Ledger application automatically calculates and creates balancing lines as required in journal entries. For example, recognizing an entity's receivable and the other entity's payable. The three balancing segment labels are primary, second, and third balancing. The primary balancing segment label is required.

Cost Center

Represents the smallest segment of an organization for which costs are collected and reported. Facilitates grouping of natural accounts by functional cost types, accommodating tracking of specific business expenses across natural accounts. As cost centers combine expenses and headcount data into costs, they are useful for detailed analysis and reporting. Cost centers are optional, but required for:

- Tracking depreciation, additions, and other transactions in Oracle Fusion Assets.
- Storing expense approval limits in Oracle Fusion Expense.

Natural Account

Determines the account type (asset, liability, expense, revenue, or equity), whether posting is allowed, and other information specific to the segment value. The natural account segment is mapped to the Financial Category dimension in the balances cube to enable ad hoc reporting and transactional dashboards. This functionality uses Oracle Fusion Business Intelligence Enterprise Edition to analyze and drill into expense and revenue transactions. The natural account segment label is required.

Intercompany

Optionally, assign the segment to be used in intercompany balancing functionality. You cannot use the primary balancing or natural account segments as the intercompany segment. You can assign the same values to both the primary balancing and intercompany value sets. You can also assign the same value set to the primary balancing segment and the intercompany segment. The sharing of the value set or values:

- Enables clear visibility of the due to and due from relationships inherent in intercompany accounting across the entire organization.
- Saves maintenance and ensure completeness.

However, sharing value sets can cause problems when applying segment value security rules as the rules applied to both segments. The rules can restrict access to certain values which may complicate entering intercompany entries. For example,



you might have access to company 01 at the balancing segment level but not company 02. As a result, you would not bet able to enter an intercompany entry for transactions between 01 and 02.

Management

Optionally, denote the segment that has management responsibility, such as the department, cost center, or line of business. Can be any segment, except the primary balancing or natural account segments. By designating a chart of accounts segment to the management segment, you can secure access to the management segment values with data access sets. By providing segment values to represent the lowest level of your organization, you can roll up results by line of business or other management criteria.

Note: Available in a future release.

Segment Labels: Example

For a chart of accounts, each segment can be qualified by a label to distinctly indicate its purpose. The label designation is used by the Oracle Fusion General Ledger processes to determine how to display and process transactions and balances that are recorded.

Scenario

You are creating your chart of accounts with six segments. Oracle Fusion General Ledger permits selection of up to thirty segments for your chart of accounts. You must have a minimum of three required segments, as determined below by the number of required segment labels (qualifiers). Required segment labels are:

- Primary Balancing Segment: Main balancing segment typically used to represent the company dimension of the organization. The segment set with this label cannot be set with another label.
- Cost Center Segment: Smallest segment of an organization for which you collect and report costs. You are required to create this segment if you are implementing Oracle Fusion Assets.
- Natural Account Segment: Classification of transactions and balances according to distinct account types: asset, liability, equity, revenue, and expense accounts. The segment set with this label cannot be set with another label.

The following optional segment labels are available and you are implementing all except for the Management Segment:

- Second Balancing Segment: Used to balance transactions, as needed, by an additional dimension beyond the primary balancing segment.
- Third Balancing Segment: Used to balance transactions, as needed, by an additional dimension beyond the primary and second balancing segments.
- Management Segment: Used in a future release. For Oracle Fusion Version 1, do not enable this qualifier.
- Intercompany Segment: Used to track intercompany due to and due from balances by identifying the specific trading company. The intercompany qualified segment cannot be set with any of the three balancing segment qualifiers. The values in this segment's value set must be the same as the primary balancing segment.

Segment labels can only be assigned once within your chart of accounts. The following table shows how you are assigning the segment labels in your chart of accounts.

Segment	Segment Label
Company	Primary Balancing Segment
Cost Center	Cost Center and Second Balancing Segment



Segment	Segment Label
Location	Third Balancing Segment
Account	Natural Account Segment
Product Line	
Intercompany	Intercompany Segment

Note: Validations are not performed when segment labels are assigned, so verify that all are assigned correctly before using your chart of accounts.

For Oracle Transactional Business Intelligence reporting, all labeled segments of the chart of accounts are automatically maintained in the data that reporting is based on. The granularity of information stored in he nonqualified segments is summarized and Oracle Transactional Business Intelligence is not able to provide detailed reporting by segments. To maintain the ability to perform detailed reporting on such segments, create custom labels to qualify these segments.

For example, one of the segments of the chart of accounts is based on product line, and none of the segment labels above are applicable. The organization must derive product line-based Oracle Transactional Business Intelligence reports. Create a custom label called Product Line to use to qualify the Product Line segment of the chart of accounts.

Balancing Segments: Explained

Balancing segments ensure that all journals balance for each balancing segment value or combination of multiple balancing segment values. You can secure access to your primary balancing segment values only with data access sets. The General Ledger application automatically calculates and creates balancing lines as required in journal entries.

The three balancing segment labels are:

- Primary
- Second
- Third
- Note: The primary balancing segment label is required.

By enabling multiple balancing segments for your chart of accounts, you can produce financial statements for each unique combination of segment values across one, two, or three qualified balancing segments. This ability provides you greater insights into your operations as it affords you visibility along the critical fiscal dimensions you use to plan, monitor, and measure your financial performance.

The following explains processes that use balancing segments.

- Intercompany balancing: Adds lines to unbalanced journals using intercompany rules.
- Opening first period of the accounting year: Calculates retained earnings amounts at the level of granularity that
 totals revenue and expense account balances for multiple balancing segment value combinations. This applies to
 standard and average balances.
- Importing journals: Adds lines using the suspense account on unbalanced journals.



- Posting journals: Adds additional lines to unbalanced journals for the following enabled account types:
 - Suspense
 - Rounding
 - Net income
 - Retained earnings
 - Cumulative translation adjustments from replication of revaluation journals to reporting currencies and for multiple reporting currency account type specific conversion
- Posting prior period journals: Calculates any income statement impact and posts to the appropriate retained earnings account.
- Translating balances: Supports multiple balancing segments for the following accounts:
 - Retained earnings: Calculated translated retained earnings are post to the retained earnings accounts by balancing segment. Retained earnings accounts represent the summing of the translated revenue and expense accounts across multiple balancing segment values.
 - Cumulative translation adjustment: Amounts posted by balancing segment to these accounts represents currency fluctuation differences between ranges of accounts which use different rate types. For example, period end rates are used for asset and liability accounts and historical rates for equity accounts.
- Revaluing Balances: Supports multiple balancing segments when calculating gain or loss accounts.
- Creating Opening Balances: Initializes reporting currency balances by converting from the total primary currency. Any difference in the reporting currency amounts is offset by populating retained earnings accounts.
- Closing year end: Supports multiple balancing segments when calculating the income statement offset and closing account in the closing journals.

Multiple Balancing Segments: Points to Consider

Oracle Fusion General Ledger supports tracking financial results at a finer level of granularity than a single balancing segment. In addition to the required primary balancing segment for the chart of accounts, which is typically associated with the company dimension of a business organization, two additional segments of the chart of accounts can be optionally qualified as the second and third balancing segments respectively. Possible chart of accounts segments that can be tagged as these additional balancing segments include cost center or department, additional aspects of a business commonly used in measuring financial results.

Several points must be consider when using multiple balancing segments:

- Journal entry processing
- Implementation timing
- Change options
- Migration adjustments

Journal Entry Processing

Multiple balancing segments ensure that account balances come from journal entries where the debits equal the credits. The financial reports are properly generated for each unique instance of account value combinations across the balancing segments. Consider this option carefully as it provides more granular reporting but requires more processing resources.



Implementation Timing

When using optional second and third balancing segments, remember that these chart of accounts segment labels are set from the beginning of time. Ensure that balances are immediately maintained in accordance with the necessary balancing actions to produce consistent financial reporting for the wanted business dimensions. Multiple balancing segment ledgers that are not maintained from the beginning of time, require extensive manual balance adjustments to catch up and realign the balances.

Note: Do not set a segment already qualified as a natural account or intercompany segment as any of the three balancing segments. Validations are not performed when segment labels are assigned, so verify that all are assigned correctly before using your chart of accounts.

Change Options

Once a segment has been enabled and designated as a balancing segment, you must not change the segment. Do not disable the segment or remove the segment labels. These settings must be consistently maintained throughout the life of the chart of accounts to control the accuracy and integrity of the financial data.

Migration Adjustments

For charts of accounts migrated from Oracle E-Business Suite to Oracle Fusion General Ledger that uses a second and third balance segments, steps must be taken to ensure the proper transition. The required adjustments are extensive.

For ledgers associated with a migrated chart of accounts, the balances must be adjusted manually. The manual adjustment is to ensure that the second and third balancing segments are consistent as though these segment labels have been in place since the beginning of entries for these ledgers. Recomputing and updating of the following processes is required to reflect the correct balancing for each account using the second and third balancing segments.

- Intercompany balancing
- Suspense posting
- Rounding imbalance adjustments on posting
- Entered currency balancing
- Revaluation gains or losses
- Retained earnings calculations at the opening of each new fiscal year
- Cumulative translation adjustments during translation

Note: All previously translated balances must also be purged. New translations must be run to properly account for translated retained earnings and cumulative translation adjustments with the correct level of balancing.

Using Multiple Balancing Segments: Example

This simple example illustrates balancing along two balancing segments for a simple chart of accounts with three segments.

Scenario

Your company has a chart of accounts with two balancing segments and three segments, qualified as follows:

- Company: Primary balancing segment
- · Cost Center: Second balancing segment
- Account: Natural account segment



The following multiple company and cost center journal transfers advertising and phone expense from Company 1, Cost Center A to Company 2, Cost Center B.

Account	Debit	Credit
Company 1-Cost Center A-Advertising Expense Account	600	
Company 2-Cost Center B-Advertising Expense Account		600
Company 1-Cost Center A-Phone Expense Account	800	
Company 2-Cost Center B-Phone Expense Account		800

During the posting process, the last four lines are created to balance the entry across the primary and second balancing segments, company and cost center.

Debit	Credit
600	
	600
800	
	800
	600
600	
	800
800	
	600



Segment Value Inheritance: Examples

The Segment Value Inheritance process simplifies the maintenance of the chart of accounts. When the characteristics of values in the value sets are updated, such as changes in enabled status, effective date, posting allowed status, or natural account type, all previously created account code combinations that referenced such values are not automatically updated by these changes. The Segment Value Inheritance process allows you to run a controlled process to update such existing account code combinations. This process maintains and corrects the current attribute settings for those account code combinations that contain the account values that were changed.

For account code combinations where the present settings need to be retained and not impacted by account attribute changes, activate the flag to preserve the account code combination's attribute. Activating the flag prevents those account code combination's attributes from being update when the Segment Value Inheritance process is run.

Scenario

For example, there are three inactive account code combinations that share a common inactive cost center value of 110.

Company-Cost Center-Account	Enabled
01-110-5210	No
04-110-4310 (Preserve Attributes flag enabled)	No
03-110-6810	No

Cost center 110 went from being disabled to enabled. When the Segment Value Inheritance process is run, the following shows the result on these three account code combinations.

Company-Cost Center-Account	Enabled
01-110-5210	Yes
04-110-4310 (Preserve Attributes flag enabled)	No
03-110-6810	Yes

Note: Once you disable a segment value and you log out of the system and back in, all code combinations containing that segment no longer work, even if the account combination still shows enabled in the account combination page. Use the Segment Value Inheritance process to set the enable flag correctly on the affected account code combinations.



FAQs for Maintain Segment Values Attributes

How can I change segments in an existing chart of accounts structure?

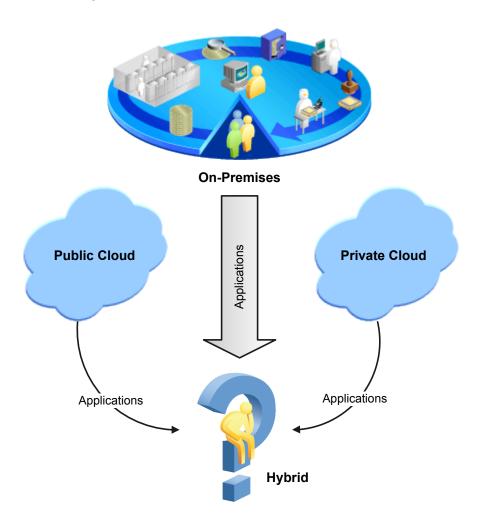
The chart of accounts structure and structure instance are fundamental constructs in General Ledger accounting setup and cannot be altered once they are used. The number of segments, the segments' order, each segment's label, length, type, and value set assignment cannot be updated. These components set the foundation upon which accounting data is recorded for ledgers that use them. Careful and thoughtful planning must precede all decisions pertaining to defining the chart of accounts.

Deploy Flexfields



Introducing the Oracle Fusion Financials Deployment Options: Overview

The following deployment options are available:



- On-Premises: Hardware, software licenses, and support that you buy and host at one or more of your sites.
- Public Cloud: Services are available to the general public and offered on a subscription basis, with no requirements
 to buy additional licenses or support. The services are hosted over the Internet by Oracle or Oracle business partners
 that offer business process outsourcing (BPO) solutions.
- Private Cloud: Application-managed and technology-managed services that are offered for Oracle software and hardware, are available internally behind a firewall, and are hosted over the Internet by Oracle or Oracle business partners that offer BPO solutions.
- **Hybrid**: Combinations you select including maintaining some or all of your existing legacy applications on-premises and buying additional products that are hosted on the private or public cloud.



Deployment Option	Key Benefits	Monitoring, Security, Patching, and Upgrades
On-Premises	You decide what you need to do with the hardware, software licenses and support you bought. You decide what to host.	You are responsible.
Public Cloud	You select services available on the public cloud on subscription basis.	Provider of the services responsible within the provisions of the subscription.
Private Cloud	Oracle provides management, monitoring, patching, security, and upgrade services.	Oracle is responsible.
Hybrid	Flexibility and lower implementation costs depending on the combination of products.	Depends on the combination.

Run Deployment: Examples

After uploading the Chart of Accounts spreadsheet, deploy the accounting flexfield

Scenario

- 1. (N) Setup and Maintenance > Deploy Chart of Accounts task > (B) Deploy the Accounting Flexfield.
- 2. (I) Refresh until the green check mark appears and verifies that the deployment is successful.
- Note: This action deploys all instances of the accounting flexfield. If there are errors in any instance the process stops. Correct the errors and run the process again.

Flexfield Deployment: Explained

Deployment generates or refreshes the Application Development Framework (ADF) business component objects that render the flexfield in a user interface. The deployment process adds custom attributes to the Web Services Description Language (WSDL) schemas exposed by Oracle ADF services and used by SOA composites. Flexfields are deployed for the first time during the application provisioning process. After you configure or change a flexfield, you must deploy it to make the latest definition available to users.

If a descriptive flexfield is enabled for business intelligence, the deployment process redeploys the flexfield's business intelligence artifacts.

You can deploy a flexfield to a sandbox for testing or to the mainline metadata for use in a test or production run time environment. You can deploy extensible flexfields as a background process.

After deployment, the custom attributes are available for incorporating into the SOA infrastructure, such as business process and business rule integration. For example, you can now write business rules that depend on the custom attributes. You must sign out and sign back in to Oracle Applications Cloud to see the changes you deployed in the run time.

The following aspects are important in understanding flexfield deployment:

- Deployment Status
- Initial Deployment Status



- Metadata Validations
- Metadata Synchronization
- Deployment as a Background Process
- Export of Artifacts from Flexfield MDS

Deployment Status

Every flexfield has a deployment status.

A flexfield can have the following deployment statuses:

Deployment Status	Meaning
Edited	The flexfield metadata definition hasn't been deployed yet. Updates of the metadata definition aren't applied in the run time environment yet.
Patched	The flexfield metadata definition has been modified through a patch or a data migration action, but the flexfield hasn't yet been deployed. So, the updated definition isn't reflected in the run time environment.
Deployed to Sandbox	The current metadata for the flexfield is deployed in ADF artifacts and available as a flexfield-enabled sandbox. The status of the sandbox is managed by the Manage Sandboxes task available to the Administrator menu of the Setup and Maintenance work area.
Deployed	The current metadata for the flexfield is deployed in ADF artifacts and available to users. No changes have been made to the flexfield after being deployed to the mainline metadata.
Error	The deployment attempt in the mainline metadata failed.

Note: Whenever a value set definition changes, the deployment status of a flexfield that uses that value set changes to edited. If the change results from a patch, the deployment status of the flexfield changes to patched.

Initial Deployment Status of Flexfields

The Oracle Applications Cloud implementation loads flexfield metadata into the database. This initial load sets the flexfield status to Edited. During installation, the application provisioning process deploys the flexfields of the provisioned applications, setting their status to Deployed if no errors occur.

In a provisioned application, deployed flexfields are ready to use. In some cases, flexfield availability at run time requires setup, such as defining key flexfields.

Metadata Validation

Use the Validate Metadata command to view possible metadata errors before attempting to deploy the flexfield. Metadata validation is the initial phase of all flexfield deployment commands. By successfully validating metadata before running the deployment commands, you can avoid failures in the metadata validation phase of a deployment attempt. The deployment process ends if an error occurs during the metadata validation phase. Metadata validation results don't affect the deployment status of a flexfield.



Metadata Synchronization

When an extensible or descriptive flexfield is deployed, the deployment process regenerates the XML schema definition (XSD). As a result, the custom attributes are available to web services and the SOA infrastructure.

After deploying a flexfield configuration, you must synchronize the updated XML schema definition (XSD) files in the MDS repositories for each SOA application.

Note: To synchronize the updated XSD files in the MDS repositories in Oracle Cloud implementations, log a service request using My Oracle Support at http://support.com/

Deployment as a Background Process

You can deploy extensible flexfields offline as a background process and continue working in the session without having to wait for the deployment to complete. You can queue up several extensible flexfields and deploy as a background process. The flexfields are deployed, one at a time, in the order that you deploy them to the queue. You must deploy extensible flexfields with more than 30 categories as a background process.

You can remove an extensible flexfield from the deployment queue with the Cancel Background Deployment command. When an extensible flexfield is deployed in a background process, its offline status indicates that the flexfield is in a background deployment process. A flexfield's offline status is cleared and its deployment status updated when the background deployment process has completed.

Export of Artifacts from Flexfield MDS

You can export business components from MDS for descriptive, extensible, or key flexfields, mainly for use in troubleshooting issues with flexfields. Use **Download Flexfield Archive** on the Manage Flexfields page to export MDS artifacts of the selected flexfield, and import them to an archive on your local computer. You can use these archived business components of flexfields for troubleshooting purposes.

Alternatively, export the deployed artifacts using exportMetadata WLST.

Related Topics

- Flexfields and Oracle Applications Cloud Architecture: How They Work Together
- Why did my flexfield changes not appear in the run time UI?

Flexfield Deployment Status: How It Is Calculated

Flexfield deployment status indicates how the flexfield metadata definition in the Oracle Fusion Applications database relates to the Application Development Framework (ADF) business components generated into an Oracle Metadata Services (MDS) Repository.

The following aspects are important in understanding how flexfield deployment status is calculated:

- Settings that affect flexfield deployment status
- How deployment status is calculated

Settings That Affect Flexfield Deployment Status

If you have made a change to a flexfield and expect a changed deployment status, be sure you have saved your changes. No settings affect flexfield deployment status.



How Deployment Status Is Calculated

If the flexfield definition has been edited through the Define Flexfields activity task flows, the status is Edited. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. Any change, including if a value set used in a flexfield changes, changes the deployment status to Edited. If a flexfield has never been deployed, its status is Edited.

Note: When an application is provisioned, the provisioning framework attempts to deploy all flexfields in that application.

If you deploy the flexfield to a sandbox successfully, the status is Deployed to Sandbox. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a sandbox MDS Repository. Whether the sandbox is active or not doesn't affect the deployment status. If the flexfield was deployed to a sandbox and hasn't been edited or redeployed to the mainline metadata since then, the status remains Deployed to Sandbox independent of whether the sandbox is active, or who is viewing the status.

If you deploy the flexfield successfully to the mainline metadata, the status is Deployed. The latest flexfield metadata definition in the Oracle Fusion application matches the metadata definition that generated ADF business components in a mainline MDS Repository. Change notifications are sent when a flexfield is deployed successfully to the mainline metadata.

If either type of deployment fails so that the current flexfield definition isn't deployed, the status is Error. The deployment error message gives details about the error. The latest flexfield metadata definition in the Oracle Fusion application likely diverges from the latest successfully deployed flexfield definition.

If the flexfield definition has been modified by a patch, the status is Patched. The latest flexfield metadata definition in the Oracle Fusion application diverges from the latest deployed flexfield definition. If the flexfield definition was Deployed before the patch and then a patch was applied, the status changes to Patched. If the flexfield definition was Edited before the patch and then a patch was applied, the status will remain at Edited to reflect that there are still changes (outside of the patch) that aren't yet in effect.

When a deployment attempt fails, you can access the Deployment Error Message for details.

Related Topics

- Flexfields and Oracle Applications Cloud Architecture: How They Work Together
- Managing Extensible Flexfields: Points to Consider

Deploying a Flexfield-Enabled Sandbox: How It Works With Mainline Metadata

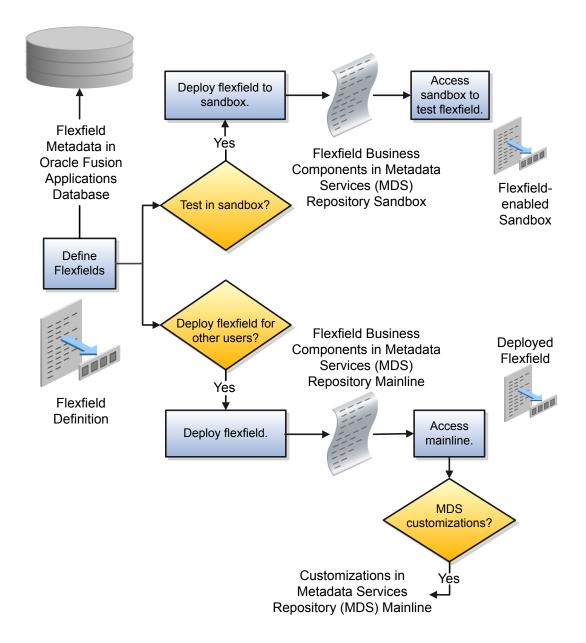
The flexfield definition in a sandbox corresponds to the flexfield metadata definition in the Oracle Fusion Applications database at the time the flexfield was deployed to the sandbox. When the flexfield is ready for end users, the flexfield must be deployed to the mainline metadata.

A flexfield-enabled sandbox uses the following components.

- Flexfield metadata in the Oracle Applications Cloud database
- Flexfield business components in a sandbox Oracle Metadata Services (MDS) repository
- User interface customizations for the flexfield in the mainline MDS repository



The figure shows the two types of deployment available in the Manage Flexfield tasks of the Define Flexfields activity. Deploying a flexfield to a sandbox creates a sandbox MDS Repository for the sole purpose of testing flexfield behavior. The sandbox is only accessible to the administrator who activates and accesses it, not to users generally. Deploying a flexfield to the mainline metadata applies the flexfield definition to the mainline MDS Repository where it is available to end users. After deploying the flexfield to the mainline metadata, customize the page where the flexfield segments appear. Customization of the page in the sandbox MDS Repository cannot be published to the mainline MDS Repository.



Sandbox Metadata Services Repository Data

Deploying the flexfield to a sandbox generates the Application Development Framework (ADF) business components of a flexfield in a sandbox MDS Repository for testing in isolation.



Caution: Don't customize flexfield segment display properties using Page Composer in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.

Mainline Metadata Services Repository Data

The Oracle Fusion Applications database stores the single source of truth about a flexfield. When the flexfield is deployed, the ADF business component objects that implement the flexfield in the run time user interface are generated in the mainline MDS Repository from this source.

Related Topics

Managing Customizations Using Sandboxes: Explained

Deploying a Flexfield to a Sandbox: Points to Consider

Deploying a flexfield to a sandbox creates a flexfield-enabled sandbox. Each flexfield-enabled sandbox contains only one flexfield.

You can test the run time behavior of a flexfield in the flexfield-enabled sandbox. If changes are needed, you return to the Define Flexfield tasks to change the flexfield definition.

When you deploy a flexfield to sandbox, the process reads the metadata about the segments from the database, generates flexfield Application Development Framework (ADF) business component artifacts based on that definition, and stores in the sandbox only the generated artifacts derived from the definition.

When you deploy a flexfield sandbox, the process generates the name of the flexfield sandbox, and that flexfield sandbox is set as your current active sandbox. When you next sign in to the application, you can see the updated flexfield configurations. The Oracle Fusion Applications global area displays your current session sandbox.

Note: Unlike a standalone sandbox created using the Manage Sandboxes tool, the sandbox deployed for a flexfield contains only the single flexfield. You can manage flexfield sandboxes, such as setting an existing flexfield sandbox as active or deleting it, using the Manage Sandboxes tool.

When you deploy a flexfield to the mainline metadata after having deployed it to the sandbox, the sandbox-enabled flexfield is automatically deleted.

Sandbox MDS Repository Data

The sandbox data lets you test the flexfield in isolation without first deploying it in the mainline metadata where it could be accessed by users.



Caution: Don't customize flexfield segment display properties using Page Composer in a flexfield-enabled sandbox as these changes will be lost when deploying the flexfield to the mainline metadata.

Managing a Flexfield-Enabled Sandbox

When you deploy a flexfield as a sandbox, that flexfield-enabled sandbox automatically gets activated in your user session. When you sign back in to see the changes, the sandbox is active in your session.

You can only deploy a flexfield to a sandbox using the Define Flexfields task flow pages.



You also can use the Manage Sandboxes feature in the Administration menu of the Setup and Maintenance work area to activate and access a flexfield-enabled sandbox.

Note: Whether you use the Define Flexfields or Manage Sandboxes task flows to access a flexfield-enabled sandbox, you must sign out and sign back in before you can see the changes you deployed in the run time.

You cannot publish the flexfield from the sandbox to the mainline metadata. You must use the Define Flexfields task flow pages to deploy the flexfield for access by users of the mainline metadata because the flexfield configuration in the mainline metadata is the single source of truth.

Related Topics

- Flexfields and Oracle Applications Cloud Architecture: How They Work Together
- Why did my flexfield changes not appear in the run time UI?
- Managing Customizations Using Sandboxes: Explained

Deploying Flexfields Using the Command Line: Explained

You can use the Manage Key Flexfields, Manage Descriptive Flexfields, and Manage Extensible Flexfields tasks to deploy flexfields. You can also use WebLogic Server Tool (WLST) commands for priming the Oracle Metadata Services (MDS) Repository with predefined flexfield artifacts and for deploying flexfields.

The table describes the available commands.

WebLogic Server Tool Command	Description
deployFlexForApp	Deploys all flexfields for the specified enterprise application. Only flexfields whose status is other than deployed are affected by this command, unless the option is enabled to force all flexfields to be deployed, regardless of deployment status. Initial application provisioning runs this command to prime the MDS Repository with flexfield
	artifacts.
deployFlex	Deploy a single flexfield regardless of deployment status
deployPatchedFlex	Deploys flexfield changes that have been delivered using a flexfield Seed Data Framework (SDF) patch. Deploys flexfields that have a Patched deployment status.
deleteFlexPatchingLabels	Displays MDS label of flexfield changes for viewing and deleting patching labels.
validateFlexDeploymentStatus	Displays list containing flexfields that aren't deployed or failed deployment.

Executing these commands outputs a report at the command line. The report provides the following information for every flexfield that is processed.

- Application identity (APPID)
- Flexfield code



Deployment result, such as success or error

In case of errors, the report lists the usages for which errors occurred. If a run time exception occurs, the output displays the trace back information. For each WLST flexfield command, adding the reportFormat='xml' argument returns the report as an XML string.

Consider the following aspects of command-line deployment.

- Preparing to use the WLST flexfield commands
- Using the deployFlexForApp command
- Using the deployFlex command
- Using the deployPatchedFlex COmmand
- Using the deleteFlexPatchingLabels Command
- Using the validateFlexDeploymentStatus Command
- Closing WLST and checking the results

Preparing To Use the WLST Flexfield Commands

You can only execute the WLST flexfield commands on a WebLogic Administration Server for a domain that has a running instance of Oracle Fusion Middleware Extensions for Oracle Application.

For more information about deploying the Oracle Fusion Middleware Extensions for Oracle Application to the server domains, see the Oracle Fusion Applications Developer's Guide.

Ensure that the AppMasterDB data source is registered as a JDBC data source with the WebLogic Administration Server and points to the same database as the ApplicationDB data source.

Start the WebLogic Server Tool (WLST) if not currently running.

UNIX:

sh \$JDEV HOME/oracle common/common/bin/wlst.sh

Windows:

wlst.cmd

Connect to the server, replacing the user name and password arguments with your WebLogic Server user name and password.

```
connect('wls_username', 'wls_password', 'wls_uri')
```

The values must be wrapped in single-quotes. The wls uri value is typically T3://localhost:7101.

For more information about the WLST scripting tool, see the Oracle Fusion Middleware Oracle WebLogic Scripting Tool.

Using the deployFlexForApp Command

The deployFlexForApp command translates the product application's predefined flexfield metadata into artifacts in the MDS Repository.



Note: This command is run automatically when you provision applications. However, if you customize applications, you have to manually run it following the order of tasks as given here:

- 1. Configure your application to read the flexfield artifacts from the MDS Repository.
- 2. Run the deployFlexForApp command.
- 3. Sign in to the application.

This sequence of steps is required even if there is no predefined flexfield metadata.

This command doesn't deploy flexfields that have a status of Deployed unless the force parameter is set to 'true' (the default setting is 'false').

For more information about priming the MDS partition with configured flexfield artifacts, see the Oracle Fusion Applications Developer's Guide.

From the WLST tool, execute the following commands to deploy the artifacts to the MDS partition, replacing product application shortname with the application's short name wrapped in single-quotes.

```
deployFlexForApp('product_application_shortname'[, 'enterprise_id'] [,'force'])
```

In a multi-tenant environment, replace enterprise id with the Enterprise ID to which the flexfield is mapped. Otherwise, replace with 'None' or don't provide a second argument.

To deploy all flexfields regardless of their deployment status, set force to 'true' (the default setting is 'false'). To deploy all flexfields in a single-tenant environment, you either can set enterprise id to 'None', or you can use the following signature:

```
deployFlexForApp(applicationShortName='product_application_shortname',force='true')
```

The application's short name is the same as the application's module name. For more information about working with application taxonomy, see the Oracle Fusion Applications Developer's Guide.

Using the deployFlex Command

From the WLST tool, execute the following command to deploy a flexfield, replacing flex code with the code that identifies the flexfield, and replacing flex type with the flexfield's type, either descriptive flexfield, key flexfield, or extensible flexfield. The values must be wrapped in single-quotes.

```
deployFlex('flex code', 'flex type')
```

Optionally, execute the following command if the flexfield is an extensible flexfield, and you want to deploy all the flexfield's configurations.

Note: By default, extensible flexfields are partially deployed. That is, only the pages, contexts, or categories that had recent changes, are deployed.

```
deployFlex('flex_code', 'flex_type', ['force_Complete_EFF_Deployment'])
where, forceCompleteEFFDeployment=None
```

Using the deployPatchedFlex Command

Use the deployPatchedFlex command for situations where the patching framework doesn't initiate the command, such as when an application has been patched offline.

If the installation is multi-tenant enabled, the command deploys all patched flexfields for all enterprises. This command isn't intended to be initiated manually.

Check with your provisioning or patching team, or the task flows for managing flexfields, to verify that the flexfield has a Patched deployment status.



From the WLST tool, execute the following command to deploy the artifacts to the MDS partition.

deployPatchedFlex()

Execute the following command to deploy all flexfields that have either a READY status or an ERROR status.

deployPatchedFlex (mode='RETRY')

Using the deleteFlexPatchingLabels Command

Whenever you deploy flexfield changes to MDS using the deployPatchedFlex() WLST command, an MDS label is created in the format FlexPatchingWatermarkdate+time. Use the deleteFlexPatchingLabels command to inquire about and delete these labels.

From the WLST tool, execute the deleteFlexPatchingLabels () command with no arguments to delete the flexfield patching labels.

To output a list of flexfield patching labels, execute the command with the infonly argument, as follows:

deleteFlexPatchingLabels(infoOnly='true')

Using the validateFlexDeploymentStatus Command

The validateFlexDeploymentStatus() WLST command checks the deployment status of all flexfields in an Oracle Fusion Applications deployment.

validateFlexDeploymentStatus()

Use this command to verify that all flexfields in the current instance of provisioned Java EE applications are deployed.

Closing WLST and Checking the Results

To close the tool, execute the command: disconnect().

Optionally, sign in the application, open user interface pages that contain flexfields, and confirm the presence of flexfields for which configuration exists, such as value sets, segments, context, or structures.

Manage Cross-Validation Rules

Cross-Validation Rules in General Ledger: Overview

You can use cross-validation rules to determine the valid account combinations that can be dynamically created as users enter transactions or journal entries. Once enabled, a cross-validation rule determines whether a selected value for a particular segment of an account combination can be combined with specific values in other segments to form a new account combination.

For example, your organization has determined that the company Operations can't use the cost center Marketing. You can define a cross-validation rule such that, if the company is Operations, then validate that the cost center isn't Marketing. New account combinations have to satisfy all of the cross-validation rules enabled for the chart of accounts before they can be created.



Entry and Maintenance

You can create cross-validation rules using the Create Cross Validation Rules in Spreadsheet task or the Manage Cross-Validations Rules task. Use the spreadsheet task to guickly enter large volumes of rules during implementation. Use the Manage Cross-Validation Rules task to add a one-off rule or edit existing rules.

To edit error messages for cross-validation rules, use the Manage Messages for General Ledger task.

Vip: When you export or import cross-validation rules to a new instance using an export or import project in the Functional Setup Manager, you must add the Manage Messages for General Ledger task before the Manage Cross-Validation Rules task. You must export or import the messages before exporting or importing the crossvalidation rules.

Existing Account Combinations

If account combinations already exist that violate newly enabled cross-validation rules, those account combinations continue to be valid. Before disabling existing account combinations that violate your rules and that you no longer use, move the balances in those accounts to the correct accounts. Then disable the account combinations manually to prevent further posting. Best practice is to define and enable cross-validation rules before: account combinations are created, transactions or journal entries are imported or entered, balances are loaded.

Related Topics

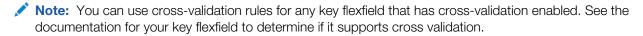
Managing Cross-Validation Rule Violations: How It Works

Cross-Validation Rules: Explained

You can control the creation of new key flexfield code combinations by defining cross-validation rules. A cross-validation rule defines validation across segments and enforces whether a value of a particular segment can be combined with specific values of other segments to form a new combination.

The table compares segment validation to cross-segment validation:

Type of validation	Type of control
Segment validation	Controls the values you can enter for a particular segment
Cross-segment validation	Controls the combinations of values that administrators and end users can create for key flexfields



Cross-validation rules prevent the creation of combinations with values that shouldn't coexist in the same combination. For example, your company requires that all revenue accounts must have a specific department. Therefore, account combinations that have revenue account values, such as all values between 4000 and 5999, must have a corresponding department value other than 000, which indicates no department is specified. You can define cross-validation rules that disallow creation of combinations with incompatible segments, such as 4100-000 or 5000-000.



Alternatively, suppose your accounting key flexfield has an Organization segment with two possible values, 01 and 02. You also have a Natural Account segment with many possible values, but company policy requires that Organization 01 uses the natural account values 001 to 499 and Organization 02 uses the natural account values 500 to 999. You can create cross-validation rules to ensure that users cannot create a general ledger account with combinations of values such as 02-342 or 01-750.

The following aspects are important to understanding cross-validation rules:

- Rule Definitions
- Enforcement
- Timing

Rule Definitions

Cross-validation rules consist of the following information:

Rule Feature	Purpose
Name	Uniquely identifies cross-validation rules in a deployment.
Description	Helps administrators identify the purpose of the rule.
Error message	Explains why the attempted combination violates the rule.
Start Date, End Date	Indicates the period of time when the rule is in effect.
Enabled	Determines whether the rule is enforced.
Condition filter	Determines the conditions under which an enabled cross-validation rule should be evaluated.
Validation filter	Determines the validation that the rule enforces when that condition is met.

When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created. If the event specified in the condition filter isn't applicable, then the combination is considered to pass the rule and the rule won't be evaluated even if it is enabled.

Note: If you don't specify any statement in the condition filter, then the condition is always true and the rule is always evaluated.

Enforcement

Cross-validation prevents creation of invalid combinations by administrators using maintenance pages and end users using dynamic insertion in foreign key pages.

Enabled rules are enforced when there is an attempt to create a new combination of segment values. Disabled rules are ignored. Deleting the rule has the same effect, but you can re-enable a disabled rule.



Timing

When users attempt to create a new combination, the key flexfield evaluates any cross-validation rules that are enabled and in effect.

Note: Cross-validation rules have no effect on combinations that already exist. The flexfield treats any existing invalid combinations that pre-date the rule as valid.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, manually disable those combinations using the combinations page for that key flexfield.

When defining a cross-validation rule, specify a start and end date to limit the time when the rule is in effect. The rule is valid for the time including the From and To dates.

Cross-Validation Rules: Points to Consider

When you need key flexfield combinations of segment values validated across segments, you can optimize your cross-validation rules to improve the experience of administrators and end users.

Consider the following when defining cross-validation rules:

- Filters
- Rule Complexity
- Maintenance

Filters

A cross-validation rule includes a condition filter and a validation filter.

The rule is evaluated using the following logic: If the condition filter is satisfied, then validate that the validation filter is satisfied.

- 1. The condition filter describes the event under which the rule will be evaluated. If the event specified in the condition filter isn't applicable, then the rule won't be evaluated even if it is enabled.
- 2. When the event specified in the condition filter is applicable, the validation filter condition must be satisfied before the combination can be created.

For example, if your organization has determined that a certain company value, Operations, cannot use a specific cost center, Marketing, you can define a cross-validation rule to validate your combinations.

- 1. The rule evaluates the company condition filter.
- 2. When company is equal to Operations, the rule evaluates the cost center validation filter.
- 3. When cost center is equal to Marketing, the rule prevents a combination from being created.
- **4.** The error message you defined for the rule displays to inform the user that the attempted combination violates the rule.
- Note: This rule doesn't affect the creation of combinations with Marketing cost center and company values other than Operations.

Rule Complexity

For optimal performance and ease of understanding, define several simple validation rules instead of using one complex rule. Simple validation rules let you provide a more specific error message and are easier to maintain over time.



Avoid rules that control validation across more than two segments, where possible. While you can define cross-validation rules that span two or more segments, keep in mind that it becomes more difficult to interpret cross-validation error messages and correct invalid key flexfield combinations as your rules encompass more segments.

Maintenance

To maintain consistent validation, review existing key flexfields when you update your cross-validation rules. Regardless of your current validation rules, Oracle Fusion Applications accept a key flexfield combination if the combination already exists and is enabled. Therefore, to ensure accurate validation, you must review your existing combinations and disable any combinations that don't match the criteria of your new rules.

▼ Tip: To keep this type of key flexfield maintenance to a minimum, decide upon your cross-validation rules when you first set up your key flexfield structure. Define cross-validation rules before creating combinations and before combinations are used in transactions.

If you want to prevent users from using previously existing combinations that are no longer valid according to your cross-validation rules, disable those combinations using the combinations page.

Editing a Cross-Validation Rule: Example

Cross-validation rules prevent specific combinations of segment values in account combinations. You can use the Manage Cross-Validation Rules task to edit existing rules or create one-off rules.

Scenario

Your organization has a cross-validation rule called Companies 131 and 151, which restricts account combinations for those companies to department 40 and product 211. Account combinations for both companies should now include department 30. To edit the cross-validation rule, perform these steps.

- 1. Navigate to the Setup and Maintenance work area. Search for and select the Manage Cross-Validation Rules task.
- 2. Select the chart of accounts for your organization and select the **Companies 131 and 151** cross-validation rule.
- 3. Click the Validation Filter icon.
- 4. Click Add Fields and select the Department segment.
- 5. Accept the default operator, which is Equals, and select department 30.
- 6. Click OK.
- 7. Click Save.
- 8. To update the error message, search for and select the Manage Messages for General Ledger task. Query the error message name for the cross-validation rule and edit the message to include department 30.

Cross-Validation Rules Spreadsheet: Explained

The rapid implementation solution provides a template for defining cross-validation rules in a spreadsheet. Cross-validation rules determine whether a selected value for a particular segment of an account combination can be combined with specific values in the other segments to form a new account combination.

You can download the spreadsheet using the Create Cross Validation Rules in a Spreadsheet task. This task is included in the following task lists:

Define Common Financials Configuration for Rapid Implementation



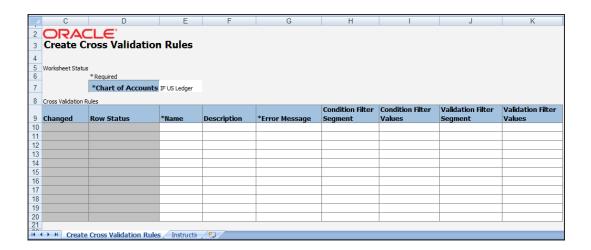
- Define Enterprise Structures Configuration for Rapid Implementation
- Note: The spreadsheet can only create cross-validation rules. To update existing cross-validation rules, use the Manage Cross-Validation Rules task.

Spreadsheet Overview

The cross-validation rules spreadsheet includes two sheets. One sheet provides the template for creating cross-validation rules and the other sheet has instructions. The Instructions sheet includes:

- An overview
- An explanation of the template
- Steps to fill in the template
- An example

The following figure shows the Create Cross Validation Rules sheet.



The following table describes each field and column on the sheet.

Field or Column	Description
Worksheet Status	The upload results for the worksheet. The application updates this field when you submit the spreadsheet.
Chart of Accounts	The chart of accounts for which the cross-validation rules are defined.
Changed	The indicator that the row has been updated. The application updates this field.
Row Status	The upload results for the row. The application updates this field when you submit the spreadsheet.
Name	The name that uniquely identifies the cross-validation rules in a deployment.
Description	The purpose for the cross-validation rule.



Field or Column	Description
Error Message	The explanation to users for why the attempted combination violates the cross-validation rule.
Condition Filter Segment	The segments of the chart of accounts that constitute the condition filter.
Condition Filter Values	The values of the condition filter segment that determine whether the cross-validation rule is evaluated.
Validation Filter Segment	The segments of the chart of accounts that constitute the validation filter.
Validation Filter Values	The values of the validation filter segment used to enforce a new account combination.

Note: The cross-validation rules created from the spreadsheet are automatically enabled and don't have a start or end date.

Steps to Use the Template

To use the spreadsheet template:

- 1. Select the chart of accounts.
- 2. Enter a suitable name, description, and error message in the respective columns.
- 3. Select the condition filter segment. To add more than one segment to the condition filter, use the next row. Repeat the rule name and select the condition filter segment.
- 4. Provide the segment values that constitute the condition filter in the Condition Filter Values column.
 - Enter detail values separated by commas for multiple detail values, for example: 5501,5502,5503.
 - Enter detail values separated by a hyphen for a range of values. You can enter multiple ranges using the comma as the range separator, for example: **3001-3030,3045-3200**.
 - Enter a parent value to select all of the detail values that are descendants of the specified parent. You can enter multiple parent values using commas as the separator, for example: **1000,2000**.
 - You could enter all of the previously listed values in the same cell, for example: 1000,2000,3001-3030,3045-3200,5501,5502,5503.
- 5. Select the validation filter segment. To add more than one segment to the validation filter, use the next row. Repeat the rule name and select the validation filter segment.
- **6.** Provide the segment values that constitute the validation filter in the **Validation Filter Values** column in the same way as specified for the condition filter.
- 7. Review the data that you entered and click **Submit** to publish the cross-validation rules.
- 8. Review the upload results in the **Worksheet Status** and **Row Status** fields.

Related Topics

• Creating Cross-Validation Rules in a Spreadsheet: Worked Example

Manage Chart of Accounts Mapping



Mapping Chart of Accounts: Explained

The chart of accounts mapping feature supports the ability to correlate a source chart of accounts to a target chart of accounts to allow for the processing of balances or amounts. Use either segment rules, account rules, or a combination of both. A chart of accounts mapping is used by the posting process in propagating transactions from the primary ledger to its secondary ledger. The mapping feature is used by both balance transfer processes for balance level secondary ledgers as well as cross ledger transfers. The balances are copied from one ledger to another ledger in both processes.

Segment Rules

Segment rules serve to map each segment of the target chart of accounts to an account value or segment in the source account. Three different mapping actions are available:

- Assign a constant value for a segment in the target chart of accounts.
- Copy the value from the source segment to the corresponding target segment.
 - Note: To use this action, the paired target and source segments must share identical values in their value sets
- Use rollup rules to aggregate source accounts to a corresponding target segment or account
 - Create a single value mapping when a specific detail source segment value is given a detail target segment value.
 - Use hierarchical rollup rules when a specific parent source value and all of its child segment values, are mapped to a given detail target segment value. This provides the ability to process groups of source segment values in one single rollup rule.
 - Define parent source values in rollup rules when date effective versions of the hierarchy are used with the accounting date of the transactions by the processes that reference the chart of accounts mapping. The additional benefit of self-maintaining mappings is that if the hierarchies referenced change with time the applicable child values are updated automatically.

Account Rules

In addition to segment rules, define account rules for the chart of accounts mapping. Account rules map a complete target account combination against one or more source account combinations. The source account combinations can be defined segment by segment using:

- Single detail account values.
- Detail account value ranges.
- Parent values for primary balancing and the natural account segments.
 - Note: When using parent values, its child values for the date effective version of the hierarchy, are processed when the mapping is called.

FAQ for Manage Chart of Accounts Mapping



What's the difference between mapping with segment rules and mapping with account rules?

Segment rules map target chart of accounts segments to an account value or segment of the source account of a secondary chart of accounts. A segment is only one part of the account combination.

Account rules map a complete target account combination against one or more source account combinations.

Note: Segment and account rules can be used alone or both types of mapping rules can be used in the same mapping set.

When do account rules override segment rules in the chart of accounts mapping?

Segment rules and account rules can be exclusively used in a chart of accounts mapping. You can use a combination of both. If there is an overlap between the two types of rule, the account rule supersedes. Segment rules are used to broadly define the relationship between two charts of accounts on a segment by segment basis. Account rules can be used to more precisely map specific source account combinations to their target accounts.

Manage DRM Synchronization

Integration with Data Relationship Management: Overview

Oracle Fusion Applications provides integration between Oracle Fusion Accounting Hub and Oracle Data Relationship Management (DRM). DRM application:

- Stores the corporate charts of accounts values and hierarchies.
- Updates the values and hierarchies to both Oracle Fusion Accounting Hub and the Oracle E-Business Suite General Ledger.

For more information on completing the postinstallation setup for Data Relationship Management, see:

- Oracle Data Relationship Management: Oracle General Ledger Integration Guide
- Oracle Data Management Integration with the Oracle Fusion and E-Business Suite General Ledgers white paper, Document ID 1315694.1, on My Oracle Support at https://support.oracle.com.

Related Topics

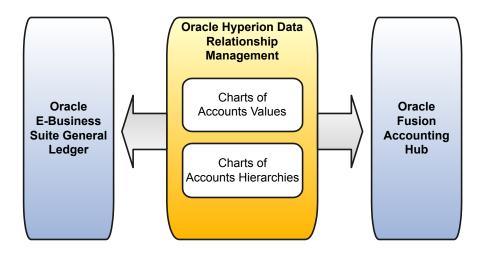
- Oracle Data Relationship Management: Oracle GL Integration Guide
- Oracle Data Relationship Management Administrator's Guide
- Oracle Data Relationship Management User's Guide
- Oracle Data Relationship Management Installation Guide



Manage Chart of Accounts Synchronization wtih DRM

Chart of Accounts Synchronization with DRM: Explained

Oracle Hyperion Data Relationship Management (DRM) is a data model-agnostic master data management solution that enables financial and analytical master data management.



- Use DRM to synchronize charts of accounts values and hierarchies across multiple Oracle E-Business Suite (EBS) Release 12 and Oracle Fusion instances.
- Use DRM to maintain a single source of truth for charts of accounts and hierarchies, to maintain charts of accounts values and hierarchies.
- Export the values to the EBS and the Oracle Fusion Accounting Hub instances.

The advantages for this process are:

- Expanded Functionality: Provides master data management for charts of accounts and hierarchies.
- Maximize Efficiency: Prevents duplication in data entry.
- Increased Accuracy: Serves as a single source of truth for all charts of accounts and hierarchies.

DRM integration with both EBS and the Oracle Fusion Accounting Hub is especially valuable if you are performing a coexistence implementation, where:

- EBS General Ledger continues to serve as the daily general ledger.
- Oracle Fusion Accounting Hub is used for financial reporting.



DRM Terminology: Explained

The following are the key terms used in Oracle Hyperion Data Relationship Management (DRM) and their usage. The terminology used in Oracle E-Business Suite (EBS) and Oracle Fusion Accounting Hub is different.

Term	Meaning	Usage
Version	Represents a single independent set of data arranged into one or more related hierarchies.	 Maps to a value set in EBS or Oracle Fusion. Maps to an Oracle Fusion Tree Version.
Hierarchy	Describe relationships between values within the Version.	Maps to EBS parent-child relationships.Maps to an Oracle Fusion Tree.
Node	A single value within a Version.	Represents chart of accounts value.
Property	Holds additional information about a version, hierarchy, or node.	 Map a Version to a value set. Map a Version to an Oracle Fusion Tree. Provide additional node information such as account category.

In DRM, the tree version refers to a single, independent set of data arranged into one or more related hierarchies.

- Similar to a value set in EBS and Oracle Fusion chart of accounts in that a version contains all the values for a segment for the chart of accounts.
- DRM usage of the tree version is not the same as an Oracle Fusion Tree Version.

Other considerations:

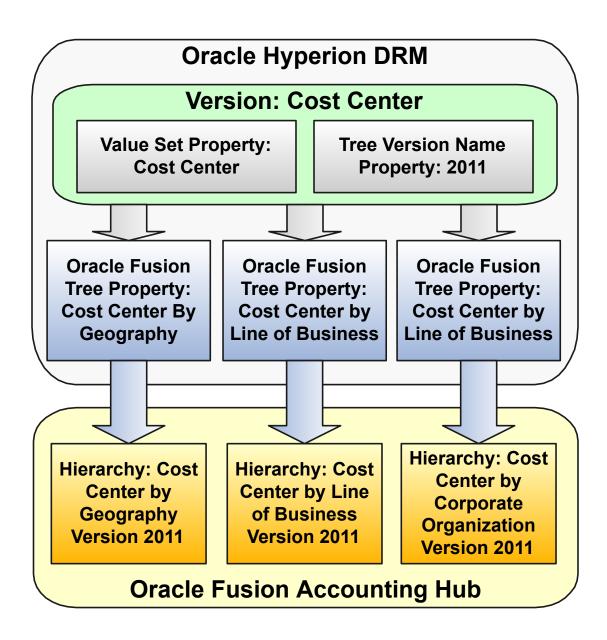
- Multiple Oracle Fusion tree versions can map to a single Oracle Fusion tree. However, only one Oracle Fusion tree
 can map to each DRM version.
- DRM hierarchy is used to store relationships between values known as nodes in DRM. These hierarchies are similar to EBS parent-child relationships, and the Oracle Fusion tree versions.
- Nodes in DRM describe a single value within a hierarchy. A node is equal to a chart of accounts value.
- Property holds a piece of information about a version, hierarchy, or node that is used to synchronize data. For
 example, there is an Oracle General Ledger property that is used to map a DRM version to a value set.

Manage Accounting Hierarchies with DRM



DRM and Oracle Fusion Accounting Hub: Examples

This diagram shows the relationships between Oracle Hyperion Data Relationship Management (DRM) versions, hierarchies, Oracle Fusion, Oracle E-Business Suite (EBS) value sets, and tree versions.



Each DRM version:

- Maps to a single segment of the chart of accounts. For example, a version can be created for 2011 and later a version for 2012 to accommodate changes in the corporate structure.
- Specifies which Oracle Fusion tree version is mapped to the DRM version.



Includes one or more DRM hierarchies.

Scenario

In this example, DRM version is created to hold values for the cost center segment. When the DRM version is created:

- The value set, called Cost Center, is mapped to the DRM version, using the Value Set Property.
- The Oracle Fusion tree version, called 2011, is mapped to the version, using the Tree Version Name Property.
 The Oracle Fusion tree version of 2011 is used to hold the Oracle Fusion tree values based upon the hierarchies associated with this version.

Scenario

In this example, three different hierarchies are associated with the version, based on the corporate desire to organize cost centers in three different ways:

- Geography
- · Line of business
- Corporate organization

For each of these hierarchies, Oracle Fusion Tree Name is specified accordingly.

Unlike Oracle Fusion, EBS does not have the ability to store date-effective hierarchies. An organization may not have a single hierarchy in EBS that represents the entire organization. For example, there can be nonrelated hierarchies for the sales organization and the manufacturing organization. Create individual hierarchies in Oracle Fusion to represent each of these hierarchies.

Individual values are stored within hierarchies. The integration program transfers those values to the Oracle Fusion Accounting Hub and EBS.

Create DRM Versions: Procedure

In Oracle Hyperion Data Relationship Management (DRM), create versions to represent your chart of account segment values and hierarchies. The versions are:

- Routed to Oracle E-Business Suite (EBS) and Oracle Fusion Accounting Hub.
- Become the single source of truth.

To create DRM versions:

- 1. Map EBS value sets to a DRM version.
- (Optional) Use the EBS Initialize Segment Values and Hierarchies process to automatically create a version for each initialized value set.
- 3. In DRM Web Client, right click on the version name and select **Properties**.
- 4. From the Category drop-down list, select Oracle General Ledger.
- 5. Complete the DRM version properties:
 - o Allow Export: Enable export to Oracle Fusion Accounting Hub and EBS.
 - **Tree Version Name**: Enter the name of **Oracle Fusion** tree that receives the hierarchy. In Oracle Fusion Applications, each DRM version is equivalent to a tree version name.
 - Value Set Master: Specify if the version is exported to EBS.



Note:

- Each value set has only one master per value set.
- The Value Set Master option is set to:
 - Yes: For a newer version.
 - No: For an older version.
- Natural Account: Specify if the value set is a natural account.
- Segment Value Max Length: Enter a value consistent with the value set configured in EBS and Oracle
 Fusion Accounting Hub. You can also create this value automatically using the import EBS initialization file.
- **6.** Use the same version to export results to multiple Oracle Fusion Applications and EBS instances. For example, define all the valid values as nodes in a single version and export it to multiple instances that use the same values in their chart of accounts.
 - Oracle Fusion Accounting Hub: Export multiple DRM hierarchies from different versions.
 - EBS: Export one DRM version per value set. In EBS, each version can have multiple hierarchies to support different requirements

Select to create as many versions in DRM as is needed.

- EBS does not store date-effective hierarchies and the existing EBS account hierarchies are overwritten with each import from DRM.
- Oracle Fusion Accounting Hub stores date-effective hierarchies, and has a synchronous relationship with DRM date-effective histories.

Initializing DRM with EBS Values: Explained

Use the Initialize Segment Values and Hierarchies program in EBS, found under the General Ledger responsibility to optionally initialize Oracle Hyperion Data Relationship Management (DRM) with Oracle E-Business Suite (EBS) values.

- The program creates a file that is used to complete the DRM import.
- Data access sets are used to determine which value sets a user may initialize.
- Each value set must be separately initialized to DRM.
- Output of this request creates a file, which you save to a location such as your desktop, to facilitate import of the file into DRM.

When you complete the import in DRM, a new DRM version exists to hold the EBS properties. If you are using DRM to synchronize with Oracle Fusion Accounting Hub, update this new DRM version to include the Oracle Fusion tree name.

Import EBS File into DRM for Initialization

After running the initialization request, use the generated output file to load the values to Oracle Hyperion Data Relationship Management (DRM).

- Select the file that you saved from the Oracle E-Business Suite (EBS) initialization program.
- Select the execute button to submit the import process.

The import process performs several actions:

Creates a DRM Version. The version name is the same as the value set name.



- Creates a node for each value in the value set, including segment properties such as whether posting to the value is allowed.
- Creates a separate hierarchy for each top node of the hierarchy in EBS. The name of each hierarchy corresponds to the description of the parent value of the top node from the parent/child relationship.
- Checks for invalid characters that are not permitted by DRM.
- Note: If you are using characters not permitted by DRM, you should:
- Fix the value set information in EBS.
- Rerun the initialization program in EBS.
- Save the output file form the initialization program.
- Rerun the import into DRM.

FAQs for Manage DRM Synchronization

How can I load EBS segment values and accounts?

Synchronization between Oracle Hyperion Data Relationship Management (DRM) and Oracle E-Business Suite (EBS) is done using a standard request called **Load Segment Values and Hierarchies** program. Submit the program in the Scheduled Processes window in EBS. Run the request for each chart of accounts segment that is to be synchronized.

The request uses data access set security to restrict which segments a user can synchronize.

When the EBS DRM import completes, an output report is generated that indicates whether the load was successful. If there are errors, there are detailed instructions for resolving them.

How can I load Oracle Fusion account values and hierarchies?

Synchronization between Oracle Fusion Accounting Hub and Oracle Hyperion Data Relationship Management (DRM) is completed by running a standard process called **Load Chart of Accounts Values and Hierarchies** from Oracle Fusion **Scheduled Processed** page.

Run the request for each value set that must be synchronized.

Define Calendars

Defining Accounting Calendars: Critical Choices

Define an accounting calendar to create your accounting year and the periods it contains. Specify common calendar options that the application uses to automatically generate a calendar with its periods. Specifying all the options makes defining a



correct calendar easier and more intuitive with fewer errors. The choices you make when specifying the following options are critical, because it is difficult to change your accounting calendar after a period status is set to open or future enterable.

- Budgetary control only
- Start Date
- Period Frequency
- Adjusting Period Frequency
- Period Name Format
- Note: In Oracle Fusion, the common calendar types, monthly, weekly, 4-4-5, 4-5-4, 5-4-4, 4-week, quarterly, and yearly, are automatically generated. This functionality makes it easier to create and maintain accounting calendars. By using the period frequency option, you no longer have to go through the tedious task of defining each period manually.

Budgetary Control Only Check Box

Select the check box for Budgetary control only to use the calendar for budgetary control only. Budgetary Control refers to the group of system options and the validation process of determining which transactions are subject to validation against budgets and budget consumption to prevent overspending.

Start Date

If you plan to run translation, specify a calendar start date that is a full year before the start date of the year of the first translation period for your ledger. Translation cannot be run in the first period of a calendar. Consider how many years of history you are going to load from your previous system and back up the start date for those years plus one more. You cannot add previous years once the first calendar period has been opened.

Period Frequency

Use period frequency to set the interval for each subsequent period to occur, for example, monthly, quarterly, or yearly. If you select the period frequency of Other, by default, the application generates the period names, year, and quarter number. You specify the start and end dates. You must manually enter the period information. For example, select the period frequency of Other and enter 52 as the number of periods when you want to define a weekly calendar. For manually entered calendars, when you click the **Add Year** button, the application creates a blank year. Then, you must manually enter the periods for the new year. The online validation helps prevent erroneous entries.

If the year has been defined and validated, use the **Add Year** button to add the next year quickly. Accept or change the new rows as required. For example, with the Other frequency type calendar, dates may differ from what the application generates.

Note: In Oracle Fusion applications a calendar can only have one period frequency and period type. Therefore, if you have an existing calendar with more than one period type associated with it, during the upgrade from Oracle E-Business Suite, separate calendars are created based on each calendar name and period type combination.

Adjusting Period Frequency

Use the adjusting period frequency to control when the application creates adjusting periods. For example, some of the frequencies you select add one adjusting period at year end, two at year end, or one at the end of each quarter. The default is None which adds no adjusting periods. If you select the frequency of Other, the **Number of Adjusting Periods** field is displayed. Enter the number of desired adjusting periods and then, manually define them.



Period Name Format Region

In the Period Name Format region enter the following fields:

- **User-Defined Prefix**: An optional feature that allows you to enter your own prefix. For example, define a weekly calendar and then enter a prefix of Week, as the separator, and the period name format of Period numberYY fiscal year. The application creates the names of Week1-11, Week2-11, through Week52-11.
- **Format**: A predefined list of values filtered on the selected separator and only displays the options that match the selected separator.
- **Year**: The year displayed in the period names is based on the selected period name format and the dates the period covers or if the period crosses years, on the year of the start date of the period.
 - For example, April 10, 2010 to May 9, 2010 has the period name of Apr-10 and December 10, 2010 to January 9, 2011 has the name of Dec-10.
 - o If period frequency is Other, then the period format region is hidden. The application generates a temporary period name for calendars with period frequency of Other, using a fixed format of Period numberYY. You can override this format with your own customized period names.
- Note: For an accounting calendar that is associated with a ledger, changing period names or adding a year updates the accounting period dimension in the balances cubes.

Calendar Validation: How It Works with the Accounting Calendar

Calendar validation is automatic and prevents serious problems when you begin using the calendar. Once you set a calendar period status to open or future enterable, you cannot edit the period.

Settings That Affect Calendar Validation

The calendar validation runs automatically when you save the calendar.

How the Calendar Is Validated

The following table lists the validation checks performed when the accounting calendar is saved.

Validation Performed	Example of Data
Unique period number	2 assigned for two periods
Unique period name	Jan-11 entered twice
Period number beyond the maximum number of periods per year	13 for a 12 period calendar with no adjusting periods
Entered period name contains spaces	Jan 11
Single or double quotes in the period name	Jan '11
Nonadjusting periods with overlapping dates	01-Jan-2011 to 31-Jan-2011 and 30-Jan-2011 to 28-Feb-2011



Validation Performed	Example of Data
Period date gaps	01-Jan-2011 to 28-Jan-2011 and 31-Jan-2011 to 28-Feb-2011
Missing period numbers	Periods 1 through 6 defined for a twelve month calendar
Period number gaps	1, 3, 5
Period numbers not in sequential order by date	Period 1 covers 01-Jan-2011 to 31-Jan-2011 and period 2 covers 01-Mar-2011 to 31-Mar-2011, and period 3 covers 01-Feb-2011 to 28-Feb-2011.
Quarter number gaps	1, 3, 4
Quarters not in sequential order by period	1, 3, 2, 4
Period start or end dates more than one year before or after the fiscal year	July 1, 2010 in a 2012 calendar

FAQs for Define Calendars

How can I identify errors in my accounting calendar?

Oracle Fusion General Ledger identifies erroneous entries online as you enter a new calendar or change data on an existing calendar. The application also automatically validates the data when you save the calendar.

What's the difference between calendar and fiscal period naming?

The period naming format determines the year that is appended to the prefix for each period in the calendar. For the example, your accounting year has a set of twelve accounting period with:

- Start date of September 1, 2014.
- End date is August 31, 2015.
- Period's date range following the natural calendar month date range.

Calendar period naming format: Select the calendar period format to append the period's start date's year to the prefix. For the period covering September 1, 2014 to December 31, 2014, then 2014 or just 14, depending on the period format selected, is appended to each period's name. For the remaining periods covering January 1, 2015 to August 31, 2015, then 2015 or 15, is appended to each period's name.

Fiscal period naming format: Select the fiscal period format to always append the period's year assignment to the prefix. If the accounting periods in the set of twelve are all assigned the year of 2015, then 2015 or just 15, depending on the period format selected, is appended to the period name of all 12 periods.



When do I update an existing calendar?

Update an existing calendar before the new periods are needed as future periods, based on the future period setting in your accounting configuration. If a complete year has been defined and validated, use the **Add Year** button to add the next year quickly. Accept or change the new rows as required. For example, with the Other frequency type calendar, dates may differ from what the application generates.

What happens if I upgrade my calendar from Oracle E-Business Suite Release 12?

The migration script assigns a period frequency that most closely matches your Oracle E-Business Suite Release 12 calendar. When you use the Oracle Fusion applications Add Year functionality for the first time, you have an opportunity to review and change the period frequency. The Calendar Options page opens only for calendars upgraded from Release 12 to allow one time modification.

Make your changes to the period frequency, adjusting period frequency, and period name format, including the prefix and separator, as needed. Changes can not conflict with the existing upgraded calendar definition. Update the calendar name and description in the calendar header, as needed, for all calendars. Period details for a new year will be generated automatically based on the latest calendar options. You can also manually update the calendar. The modified calendar options affect future years only.

Manage Currencies

Defining Currencies: Points to Consider

When creating or editing currencies, consider these points relevant to entering the currency code, date range, or symbol for the currency.

Currency Codes

You can't change a currency code after you enable the currency, even if you later disable that currency.

Date Ranges

You can enter transactions denominated in the currency only for the dates within the specified range. If you don't enter a start date, then the currency is valid immediately. If you don't enter an end date, then the currency is valid indefinitely.

Symbols

Some applications support displaying currency symbols. You may enter the symbol associated with a currency so that it appears along with the amount.



Euro Currency Derivation: Explained

Use the Derivation Type, Derivation Factor, and Derivation Effective Date fields to define the relationship between the official currency (Euro) of the European Monetary Union (EMU) and the national currencies of EMU member states. For each EMU currency, you define its Euro-to-EMU fixed conversion rate and the effective starting date. If you have to use a different currency for Euro, you can disable the predefined currency and create a new one.

Derivation Type

The **Euro currency** derivation type is used only for the Euro, and the **Euro derived** derivation type identifies national currencies of EMU member states. All other currencies don't have derivation types.

Derivation Factor

The derivation factor is the fixed conversion rate by which you multiply one Euro to derive the equivalent EMU currency amount. The Euro currency itself must not have a derivation factor.

Derivation Effective Date

The derivation effective date is the date on which the relationship between the EMU currency and the Euro begins.

FAQs for Manage Currencies

When do I create or enable currencies?

Create or enable any currency for displaying monetary amounts, assigning currency to ledgers, entering transactions, recording balances, or for any reporting purpose. All currencies listed in the International Organization for Standardization (ISO) 4217 standard are supported.

The default currency is set to United States Dollar (USD).

What's the difference between precision, extended precision, and minimum accountable unit for a currency?

Precision refers to the number of digits placed to the right of the decimal point used in regular currency transactions. For example, USD would have 2 as the precision value for transactional amounts, such as \$1.00.

Extended precision is the number of digits placed to the right of the decimal point and must be greater than or equal to the precision value. For calculations requiring greater precision, you can enter an extended precision value such as 3 or 4. That would result in the currency appearing as \$1.279 or \$1.2793.

Minimum accountable unit is the smallest denomination for the currency. For example, for USD that would be .01 for a cent.

In Setup and Maintenance work area, search for the Manage Currencies task to set these values for a currency.



Manage Conversion Rate Types

Creating Conversion Rate Types: Critical Choices

Maintain different conversion rates between currencies for the same period with the Oracle Fusion General Ledger conversion rate types functionality. Four predefined daily conversion rate types are seeded:

- Spot
- Corporate
- User
- Fixed

You can use different rate types for different business needs. During journal entry, the conversion rate is provided automatically by the General Ledger based on the selected conversion rate type and currency, unless the rate type is user. For user rate types, you must enter the conversion rate. Define additional rate types as needed. Set your most frequently used rate type as the default. Conversion rate types cannot be deleted.

Assign conversion rate types to automatically populate the associated rate for your period average and period end rates for the ledger. For example, you can assign the predefined rate type **Spot** to populate your period average rates and the predefined rate type **Corporate** to populate your period end rates. Period average and period end rates are used in translation of account balances.

Conversion rate types are used to automatically assign a rate when you perform the following accounting functions:

- Convert foreign currency journal amounts to ledger currency equivalents.
- Convert journal amounts from source ledgers to reporting currencies or secondary ledgers.
- Run Revaluation or Translation processes.

In creating new conversion rates, decide whether to do the following:

- Enforce inverse relationships
- Select pivot currencies
- Select contra currencies
- Enable cross rates and allow cross rate overrides
- Maintain cross rate rules

Enforce Inverse Relationships

Select the **Enforce Inverse Relationship** option to specify whether or not to enforce the automatic calculation of inverse conversion rates when defining daily rates.

Action	Results
Selected	When you enter a daily rate to convert currency A to currency B, General Ledger automatically calculates the inverse rate, currency B to A, and enters it in the adjacent column. If either rate is changed, the application automatically recalculates the other rate.



Action	Results You can update the application calculated inverse rate, but once you do, the related rate is updated. The check box enforces that the inverse relationship is maintained but does not prevent you from updating the rates.
Not Selected	General Ledger calculates the inverse rate but you can change the rate and update the daily rates table without the corresponding rate being updated.

Select Pivot Currencies

Select a pivot currency that is commonly used in your currency conversions. A pivot currency is the central currency that interacts with contra currencies. For example, you set up a daily rate between the US dollar (USD) and the Euro currency (EUR) and another between the USD and the Canadian dollar (CAD). USD is the pivot currency in creating a rate between EUR and CAD. EUR and CAD are the contra currencies. Select the pivot currency from the list of values which contains those currencies that are enabled, effective, and not a statistical (STAT) currency. The description of the pivot currency is populated automatically based on the currency definition.

If you want the application to create cross rates against a base currency, define the base currency as the pivot currency. Selected pivot currencies can be changed in the Rate Types page.

Select Contra Currencies

Select currencies available on the list of values as contra currencies. The available currencies are those currencies which are enabled, effective, not STAT currency, and not the pivot currency selected earlier. The description of the contra currency is populated automatically based on the currency definition. Add or delete contra currencies in the Contra Currencies region of the Rate Types page.

Enable Cross Rates and Allow Cross Rate Overrides

Check the **Enable Cross Rates** check box to calculate conversion rates based on defined currency rate relationships. General Ledger calculates cross rates based on your defined cross rate rules. Associate your cross rate rules with a conversion rate type, pivot currency, and contra currencies. Cross rates facilitate the creation of daily rates by automatically creating the rates between contra currencies based on their relationship to a pivot currency. If the **Enable Cross Rates** check box is changed to unchecked after entering contra currencies, the application stops calculating cross rates going forward for that particular rate type. All the earlier calculated cross rates for that rate type remain in the database unless you manually delete them.

For example, if you have daily rates defined for the pivot currency, USD to the contra currency, EUR, and USD to another contra currency, CAD, the application will automatically create the rates between EUR to CAD and CAD to EUR. This prevents the need to manually define the EUR to CAD and CAD to EUR rates.

Check the **Allow Cross Rates Override** check box to permit your users to override application generated cross rates. If you accept the default of unchecked, the application generated cross rates cannot be overridden

Maintain Cross Rate Rules

Define or update your cross rate rules at any time by adding or removing contra currency assignments. Add a contra currency to a cross rate rule and run the Daily Rates Import and Calculation process to generate the new rates. If your remove a cross rate rule or a contra currency from a rule, any cross rates generated previously for that contra currency remain unless you manually delete them. Changes to the rule are not retroactive and will not affect previously stored cross rates. The Cross Rate process generates as many rates as possible and skips currencies where one component of the set is missing.



Note: With a defined web service that extracts daily currency conversion rates from external services, for example Reuters, currency conversion rates are automatically updated for the daily rates and all cross currency relationships.

Using Rate Types: Examples

The four predefined conversion rate types in Oracle Fusion Applications are:

- Spot
- Corporate
- User
- Fixed

Scenario

You are the general ledger accountant for Vision US Inc. You are entering a journal entry to capture three transactions that were transacted in three different foreign currencies:

- Canadian dollar (CAD): A stable currency
- Mexican Peso (MXP): A fluctuating currency
- Hong Kong dollar (HKD): An infrequently used currency

You enter two lines with accounts and amounts for each foreign currency transaction. Based on your company procedures, you select the rate type to populate the rate for **Corporate** and **Spot** rate types from your daily rates table. You manually enter the current rate for the **User** rate type.

Currency Selected	Rate Type Selected	Reason
CAD	Corporate	Entered a periodic type of transaction. Your company has established a daily rate to use for the entire month across divisions for all transactions in CAD. CAD is a stable currency that only fluctuations slightly over the month.
MXP	Spot	Entered a periodic type of transaction. Your company enters daily rates each day for MXP because this currency is unstable and fluctuates.
HKD	User	Entered a one time transaction. Your company does not maintain daily rates in HKD.

Your company does not currently use the **Fixed** rate type. From January 1, 1999, the conversion rate of the French franc (FRF) against the Euro (EUR) was a fixed rate of 1 EUR to 6.55957 FRF. Your French operations were started in 2007, so you maintain all your French business records in the EUR.



FAQs for Manage Conversion Rate Types

What's the difference between spot, corporate, user, and fixed rate types?

Spot, corporate, user, and fixed conversion rate types differ based on fluctuations of your entered foreign currency and your company procedures for maintaining daily rates.

Rate Type	Usage
Spot	For currencies with fluctuating conversion rates or when exact currency conversion is needed.
Corporate	For setting a standard rate across your organization for a stable currency.
User	For infrequent entries where your daily rates for the entered foreign currency are not set up.
Fixed	For rates where the conversion is constant between two currencies.

If you have infrequent foreign currency transactions, the user rate type can simplify your currency maintenance. The user rate can also provide an accurate conversion rate on the date of the transaction.

What's a statistical unit currency type?

The statistical unit currency type denotes the Statistical (STAT) currency used to record financial statistics in the financial reports, allocation formulas, and other calculations.

Manage Daily Rates

Load GL Currency Rates: Overview

Oracle Cloud ERP has various options to load currency rates. In Oracle Cloud ERP, daily currency conversion rates can be maintained between any two currencies. You can enter daily conversion rates for specific combinations of foreign currency, date, and conversion rate type. Oracle Cloud ERP automatically calculates inverse rates. You can override the inverse rates that are automatically calculated.

The three different methods of loading currency rates are:

- 1. Manual load using the **Daily Rates** spreadsheet
- Manual load using File Based Data Import process.
- 3. Automatic load using web services.



Note: If you encounter errors loading currency rates, you can correct the errors and reload.

Entering Daily Rates Manually: Worked Example

You are required to enter the daily rates for currency conversion from Great Britain pounds sterling (GBP) to United States dollars (USD) for 5 days for your company InFusion America Inc.

In order to load rates using the Daily Rates Spreadsheet, you need to install Oracle ADF Desktop Integration client software. Oracle ADF Desktop Integration is an Excel add-in that enables desktop integration with Microsoft Excel workbooks. Users can download the installation files from **Navigator > Tools > Download Desktop Integrator Installer.**

Entering Daily Rates

- 1. Navigator > Period Close.
 - Use the **Period Close** work area to link to close processes and currency process.
- 2. Click the Manage Currency Rates link.
 - Use the **Currency Rates Manager** page to create, edit, and review currency rate types, daily rates, and historical rates.
- 3. Click the **Daily Rates** tab.
 - Use the **Daily Rates** tab to review and enter currency rates.
- 4. Click the **Create in Spreadsheet** button.
 - Use the Create Daily Rates spreadsheet to enter daily rates in a template that you can save and reuse.
- **5.** Click in the **From Currency** field. Select the GBP Pound Sterling list item.
- **6.** Click in the **To Currency** field. Select the USD US Dollar list item.
- 7. Click in the Conversion Rate field. Select the Spot list item
- 8. Click in the **From Conversion** field. Enter a valid value: 10/1/2014.
- 9. Click in the **To Conversion Date** field. Enter a valid value: 10/5/2014.
- 10. Click in the Conversion Rate field. Enter a valid value: 1.6.
- **11.** Click the **Submit** > **OK** twice.
- 12. Review the **Record Status** column to verify that all rows were loaded successfully.
- 13. Save template to use to enter daily rates frequently. You can save the spreadsheet to either a local drive or a shared network drive.
- 14. Optionally, edit the rates from the Daily Rates user interface or resubmit the spreadsheet.

Related Topics

Using Desktop Integrated Excel Workbooks: Points to Consider

GL Currency Rates File Based Load: Examples

Oracle File Based Data Import (FBDI) is used when you must load high volume of daily rates data to Oracle Cloud ERP.

Scenario

You are assigned the task to upload daily rates for your company. You have decided to use the File Based Data Import and perform the following steps.



- Note: For more information about file-based data import, see the File Based Data Import for Oracle Financials Cloud guide.
- 1. Download the Daily Rates template called DailyRatesImportTemplate.xlsm from the File Based Data Import for Oracle Financials Cloud guide.
- 2. Enter the rates data in the FBDI template. To prepare your data in a spreadsheet format, use XLS templates.
 - a. The first worksheet in each template provides instructions for using the template.
 - **b.** Save a copy of the file.
 - **c.** Enable macros using **Options** > **Enable** this content radio button.
- 3. Validate data.

You can use template design to perform extensive validations at the preliminary level.

- Checks data formats and other field checks. if a rate overlap is present for a particular combination of From and To Currency code and rate type. Errors are identified and presented in a separate sheet for the users to review and correct
- Checks if a rate overlap is present for a particular combination of From and To Currency codes and rate type.
- Identifies errors and presents them in a separate sheet for the users to review and correct them before revalidating.
- 4. Generate daily rates .zip file.

Generate daily rates .zip file containing the validated and formatted comma-separated values (.csv) file by selecting **Generate Daily Rates** button within the template.

- 5. Upload file in Universal Content Management (UCM) system which is a single platform for managing contents across your enterprise.
 - a. Use Navigator > File Import and Export to upload the file in the UCM server.
 - **b.** Use the **Account** as fin/generalLedger/import when uploading.
 - c. Save and Close.
- 6. Run Load Interface File for Import Process from Navigator -> Tools-> Scheduled processes->Schedule New Process. This process:
 - Validates the data before uploading the data to the GL Daily Rates Interface table.
 - **b.** Allows review the error log and correction of errors before running the process again.
 - c. Populate GL Daily Rates table automatically as a post import process once data is successfully validated.
- 7. Review the imported rates by navigating to **Manage Currency Rates** page.
- Note: For more information see: Load Currency Rates To Oracle Fusion ERP Cloud (Doc ID 1950660.1) on My Oracle Support.

Related Topics

• File Based Data Import for Oracle Financials Cloud



Automatic Load Using Web Services: Examples

If you have a rate subscription source that provides daily rates, use a web service to fully automate the loading of the rates using the **File Based Data Import**. The external web service **FinancialUtilService** enables you to upload data files to the Oracle Universal Content Management (UCM), and launch the processes to import the rates to Oracle Cloud ERP.

Scenario

You are setting up an automatic process to load your company's daily rates.

Process Steps:

- 1. Get rates file from the designated external source.
 - If you have a rate subscription source that provides daily rates, then you can call web service to collect the file and to feed into Oracle Data Integrator (ODI) or alternatively to a temporary storage in the UCM that in turn can be fed into ODI for further processing.
- 2. Prepare and validate data.
 - You would need to perform data transformation to fit the format of the spreadsheet template. These data processing steps can be accomplished using the ability extended by Oracle Data Integrator (ODI).
 - The .csv data file needs to be provided and transformed in the specific format, in the exact column sequence as in the .xlsm file and data types as expected. For example, the .csv file provides columns such as from currency code, to currency code, dates in canonical format, conversion rate type, and the rate.
- 3. Call web service uploadFileToUcm.
 - Automatically upload the formatted, validated file from ODI into the Universal Content Management (UCM).
- 4. Call web service submitESSJobRequest.
 - Submit the Load Interface File for Import process to load data into GL Daily Rates interface table.
- 5. Call web service **getESSJobStatus**.
 - Verify the status of request submitted in previous step. If the process ended in errors, you may need to correct them and then resubmit the process.
- **6.** The process to submit the import rates from interface table to **GL Daily Rates** table happens automatically as a post process to the submit the **Load Interface File for Import** process performed in step four.
- 7. Once the **Import and Calculate Daily Rates** process completes successfully, you can verify and review the rates from the **Currency Rates Manager** page.

Security Required for Loading GL Currency Rates: Explained

When load rates to Oracle Cloud ERP, you must be assigned the following security privileges.

Tasks	Privilege	Duties
Creating rate types	Define Conversion Rate Type	Daily Rates Administration Duty
		General Accounting Manager
Maintaining daily conversion rates	Maintain Daily Conversion Rate	Daily Rates Administration Duty



Tasks	Privilege	Duties	
		General Accounting Manager	
Importing rate to Cloud ERP	Run Daily Rates Import and Calculation Process	Daily Rates Administration Duty	
	FIOCESS	General Accounting Manager	
Viewing daily conversion rates	View Daily Conversion Rate	Daily Rates Administration Duty	
		General Accounting Setup Review Duty	

Updating Currency Rates: Worked Example

You are required to change today's daily rates that were already entered. The rates you are changing are for currency conversion from Great Britain pounds sterling (GBP) to United States dollars (USD) for your company InFusion America.

Currency conversion rates were entered by an automatic load to the Daily Rates table. They can also be entered through a spreadsheet.

Updating Currency Rates

1. Navigate to the Period Close work area.

Use the Period Close work area to link to close processes and currency process.

2. Click the Manage Currency Rates link.

Use the Currency Rates Manager page to create, edit, and review currency rate types, daily rates, and historical rates.

3. Click the Daily Rates tab.

Use the Daily Rates tab to review and enter currency rates.

- 4. Click the From Currency list. Select the GBP Pound Sterling list item.
- 5. Click the To Currency list. Select the USD US Dollar list item.
- 6. Enter the dates for the daily rates that you are changing. Enter today's date.
- 7. Click the Rate Type list. Select the Spot list item.
- 8. Click the Search button.
- 9. Click in the Rate field. Enter the new rate of 1.7 in the Rate field.
- 10. Click in the Inverse Rate field. Enter the new inverse rate of 0.58822 in the Inverse Rate field.
- 11. Click the Save button.

Related Topics

Using Desktop Integrated Excel Workbooks: Points to Consider



4 Define Ledgers

Ledgers and Subledgers: Explained

Oracle Fusion Applications reflect the traditional segregation between the general ledger and associated subledgers. Detailed transactional information is captured in the subledgers and periodically imported and posted in summary or detail to the ledger.

A ledger determines the currency, chart of accounts, accounting calendar, ledger processing options, and accounting method for its associated subledgers. Each accounting setup requires a primary ledger and optionally, one or more secondary ledgers and reporting currencies. Reporting currencies are associated with either a primary or secondary ledger.

The number of ledgers and subledgers is unlimited and determined by your business structure and reporting requirements.

Single Ledger

If your subsidiaries all share the same ledger with the parent company or they share the same chart of accounts and calendar, and all reside on the same applications instance, you can consolidate financial results in Oracle Fusion General Ledger in a single ledger. Use Oracle Fusion Financial Reporting functionality to produce individual entity reports by balancing segments. General Ledger has three balancing segments that can be combined to provide detailed reporting for each legal entity and then rolled up to provide consolidated financial statements.

Multiple Ledgers

Accounting operations using multiple ledgers can include single or multiple applications instances. You need multiple ledgers if one of the following is true:

- You have companies that require different account structures to record information about transactions and balances.
 For example, one company may require a six-segment account, while another needs only a three-segment account structure.
- You have companies that use different accounting calendars. For example, although companies may share fiscal
 year calendars, your retail operations require a weekly calendar, and a monthly calendar is required for your
 corporate headquarters.
- You have companies that require different functional currencies. Consider the business activities and reporting
 requirements of each company. If you must present financial statements in another country and currency, consider
 the accounting principles to which you must adhere.

Subledgers

Oracle Fusion Subledgers capture detailed transactional information, such as supplier invoices, customer payments, and asset acquisitions. Oracle Fusion Subledger Accounting is an open and flexible application that defines the accounting rules, generates detailed journal entries for these subledger transactions, and posts these entries to the general ledger with flexible summarization options to provide a clear audit trail.



Ledgers: Points to Consider

Companies account for themselves in primary ledgers, and, if necessary, secondary ledgers and reporting currencies. Transactions from your subledgers are posted to your primary ledgers and possibly, secondary ledgers or reporting currencies based on balance, subledger, or journal level settings. Local and corporate compliance can be achieved through an optional secondary ledger. Provide an alternate accounting method, or in some cases, a different chart of accounts. Your subsidiary's primary and secondary ledgers can both be maintained in your local currency. You can convert your local currency to your parent's ledger currency to report your consolidated financial results using reporting currencies or translation.

Primary Ledgers

A primary ledger:

- Is the main record-keeping ledger.
- Records transactional balances by using a chart of accounts with a consistent calendar and currency, and accounting rules implemented in an accounting method..
- Is closely associated with the subledger transactions and provides context and accounting for them.

To determine the number of primary ledgers, your enterprise structure analysis must begin with your financial, legal, and management reporting requirements. For example, if your company has separate subsidiaries in several countries worldwide, enable reporting for each country's legal authorities by creating multiple primary ledgers that represent each country with the local currency, chart of accounts, calendar, and accounting method. Use reporting currencies linked to your country-specific primary ledgers to report to your parent company from your foreign subsidiaries. Other considerations that affect the number of primary ledgers required are:

- Corporate year end
- Ownership percentages
- Local government regulations and taxation
- Secondary ledgers

Secondary Ledgers

A secondary ledger:

- Is an optional ledger linked to a primary ledger for the purpose of tracking alternative accounting.
- Can differ from its primary ledger by using a different accounting method, chart of accounts, accounting calendar, currency, or processing options.

When you set up a secondary ledger using the Manage Secondary Ledger task, you select a data conversion level. The data conversion level determines what level of information is copied to the secondary ledger. You can select one of the following levels: **Balance**, **Journal**, **Subledger**, or **Adjustment Only**.

- **Balance**: When you run the Transfer Balances to Secondary Ledger process, balances are transferred from the primary ledger to the secondary ledger.
- **Journal**: When you post journals in the primary ledger, the posting process copies the journals to the secondary ledger for the sources and categories that you specify in the Journal Conversion Rules section on the Map Primary to Secondary Ledger page.



In the Journal Conversion Rules section, you can do one of the following:

- Accept the default setting of Yes for the journal source and category combination of Other, and then specify
 the source and category combinations to exclude from the conversion.
- Set the journal source and category combination of **Other** to **No**, and then specify the source and category combinations to include in the conversion.
- **Subledger**: When you generate accounting in the primary ledger, the Create Accounting process produces accounting for both the primary and secondary ledgers. When you post journals in the primary ledger, the posting process copies the journals to the secondary ledger for the sources and categories that you specify in the Journal Conversion Rules section on the Map Primary to Secondary Ledger page.
 - ▲ Caution: Ensure you have journal conversion rules that exclude your subledgers, otherwise your subledger data is copied to the secondary ledger twice, once by the Create Accounting process, and once by posting.
- Adjustment Only: This level is an incomplete accounting representation that holds only adjustments. The
 adjustments can be manual adjustments or automated adjustments from subledger accounting. This type of
 secondary ledger must share the same chart of accounts, accounting calendar, period type, and currency as the
 associated primary ledger.
- Tip: To obtain a complete secondary accounting representation that includes both transactional data and adjustments, use ledger sets to combine the ledgers when running reports.

Example

Your primary ledger uses US Generally Accepted Accounting Principles (GAAP) and you maintain a secondary ledger for International Financial Reporting Standards (IFRS) accounting requirements. You first decide to use the subledger conversion level for the IFRS secondary ledger. However, since most of the accounting between US GAAP and IFRS is identical, the adjustment only level is the better solution for the secondary ledger. The subledger level requires duplication of most subledger and general ledger journal entries and general ledger balances. The adjustment only level transfers only the adjustment journal entries and balances necessary to convert your US GAAP accounting to the IFRS accounting. Thus, requiring less processing resources.

Tip: To avoid difficult reconciliations, use the same currency for primary and secondary ledgers. Use reporting currencies or translations to generate the different currency views to comply with internal reporting needs and consolidations.

Reporting Currencies

Reporting currencies maintain and report accounting transactions in additional currencies. Consider the following before deciding to use reporting currencies.

- Each primary and secondary ledger is defined with a ledger currency that is used to record your business transactions and accounting data for that ledger.
- Best practices recommend that you maintain the ledger in the currency in which the majority of its transactions are denominated. For example, create, record, and close a transaction in the same currency to save processing and reconciliation time.
- Compliance, such as paying local transaction taxes, is also easier using a local currency.



Many countries require that your accounting records be kept in their national currency.

If you maintain and report accounting records in different currencies, you do this by defining one or more reporting currencies for the ledger. When you set up a reporting currency using the Manage Reporting Currency task, you select a currency conversion level. The currency conversion level determines what level of information is copied to the reporting currency.

You can select one of the following levels: Balance, Journal, Subledger.

- **Balance**: When you run the Translate General Ledger Account Balances process, balances are transferred from the specified ledger to the reporting currency and converted.
- **Journal**: When you post journals, the posting process copies the journals to the reporting currency for the sources and categories that you specify in the Journal Conversion Rules section on the Create or Edit Reporting Currency pages.

In the Journal Conversion Rules section, you can do one of the following:

- Accept the default setting of Yes for the journal source and category combination of Other, and then specify
 the source and category combinations to exclude from the conversion.
- Set the journal source and category combination of **Other** to **No**, and then specify the source and category combinations to include in the conversion.
- **Subledger**: When you generate accounting in the primary ledger, the Create Accounting process produces accounting for both the primary ledger and the reporting currency. When you post the journals in the primary ledger, the posting process copies the journals to the reporting currency for the sources and categories that you specify in the Journal Conversion Rules section on the Create or Edit Reporting Currency pages.
 - ▲ Caution: Ensure you have journal conversion rules that exclude your subledgers, otherwise your subledger data is copied to the reporting currency twice, once by the Create Accounting process, and once by posting.
- Note: A full accounting representation of your primary ledger is maintained in any subledger level reporting currency. Secondary ledgers cannot use subledger level reporting currencies.

Do not use journal or subledger level reporting currencies if your organization translates your financial statements to your parent company's currency for consolidation purposes infrequently. Standard translation functionality meets this need. Consider using journal or subledger level reporting currencies when any of the following conditions exist.

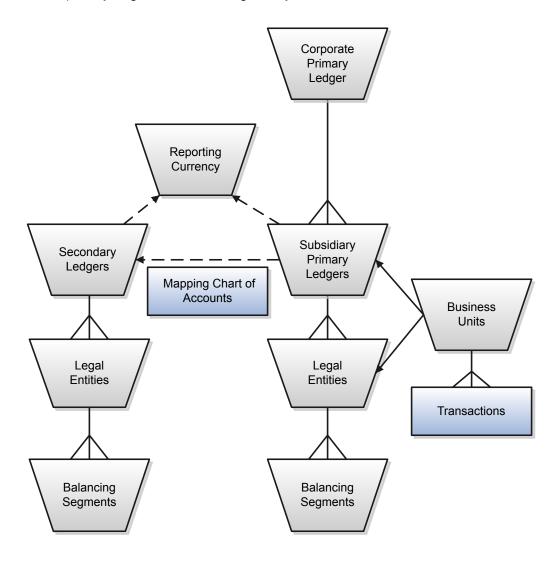
- You operate in a country whose unstable currency makes it unsuitable for managing your business. As a
 consequence, you manage your business in a more stable currency while retaining the ability to report in the
 unstable local currency.
- You operate in a country that is part of the European Economic and Monetary Union (EMU), and you select to account and report in both the European Union currency and your National Currency Unit (NCU).
- Note: The second option is rare since most companies have moved beyond the initial conversion to the EMU currency. However, future decisions could add other countries to the EMU, and then, this option would again be used during the conversion stage.



Financial Ledgers: How They Fit Together

Oracle Fusion Applications is an integrated suite of business applications that connects and automates the entire flow of the business process across both front and back office operations and addresses the needs of a global enterprise. The process of designing the enterprise structure, including the accounting configuration, is the starting point for an implementation. This process often includes determining financial, legal, and management reporting requirements, setting up primary and secondary ledgers, making currency choices, and examining consolidation considerations.

This figure shows the enterprise structure components and their relationships to each other. Primary ledgers are connected to reporting currencies and secondary ledgers to provide complete reporting options. Legal entities are assigned to ledgers, both primary and secondary, and balancing segments are assigned to legal entities. Business units must be connected to both a primary ledger and a default legal entity. Business units can record transactions across legal entities.





Primary Ledgers

A primary ledger is the main record-keeping ledger. Create a primary ledger by combining a chart of accounts, accounting calendar, ledger currency, and accounting method. To determine the number of primary ledgers, your enterprise structure analysis must begin with determining financial, legal, and management reporting requirements. For example, if your company has separate subsidiaries in several countries worldwide, create multiple primary ledgers representing each country with the local currency, chart of accounts, calendar, and accounting method to enable reporting to each country's legal authorities.

If your company just has sales in different countries, with all results being managed by the corporate headquarters, create one primary ledger with multiple balancing segment values to represent each legal entity. Use secondary ledgers or reporting currencies to meet your local reporting requirements, as needed. Limiting the number of primary ledgers simplifies reporting because consolidation is not required. Other consideration such as corporate year end, ownership considerations, and local government regulations, also affect the number of primary ledgers required.

Secondary Ledgers

A secondary ledger is an optional ledger linked to a primary ledger. A secondary ledger can differ from its related primary ledger in chart of accounts, accounting calendar, currency, accounting method, or ledger processing options. Reporting requirements, for example, that require a different accounting representation to comply with international or country-specific regulations, create the need for a secondary ledger.

Below are scenarios and required action for different components in primary and secondary ledgers:

- If the primary and secondary ledgers use different charts of accounts, the chart of accounts mapping is required to instruct the system how to propagate journals from the source chart of accounts to the target chart of accounts.
- If the primary and secondary ledgers use different accounting calendars, the accounting date and the general ledger
 date mapping table will be used to determine the corresponding non-adjusting period in the secondary ledger. The
 date mapping table also provides the correlation between dates and non-adjusting periods for each accounting
 calendar.
- If the primary ledger and secondary ledger use different ledger currencies, currency conversion rules are required to
 instruct the system on how to convert the transactions, journals, or balances from the source representation to the
 secondary ledger.

Note: Journal conversion rules, based on the journal source and category, are required to provide instructions on how to propagate journals and types of journals from the source ledger to the secondary ledger.

Reporting Currencies

Reporting currencies are the currency you use for financial, legal, and management reporting. If your reporting currency is not the same as your ledger currency, you can use the foreign currency translation process or reporting currencies functionality to convert your ledger account balances in your reporting currency. Currency conversion rules are required to instruct the system on how to convert the transactions, journals, or balances from the source representation to the reporting currency.

Legal Entities

Legal entities are discrete business units characterized by the legal environment in which they operate. The legal environment dictates how the legal entity should perform its financial, legal, and management reporting. Legal entities generally have the right to own property and the obligation to comply with labor laws for their country. They also have the responsibility to account for themselves and present financial statements and reports to company regulators, taxation authorities, and other



stakeholders according to rules specified in the relevant legislation and applicable accounting standards. During setup, legal entities are assigned to the accounting configuration, which includes all ledgers, primary and secondary.

Balancing Segments

You assign primary balancing segment values to all legal entities before assigning values to the ledger. Then, assign specific primary balancing segment values to the primary and secondary ledgers to represent nonlegal entity related transactions such as adjustments. You can assign any primary balancing segment value that has not already been assigned to a legal entity. You are allowed to assign the same primary balancing segment values to more than one ledger. The assignment of primary balancing segment values to legal entities and ledgers is performed within the context of a single accounting setup. The Balancing Segment Value Assignments report is available to show all primary balancing segment values assigned to legal entities and ledgers across accounting setups to ensure the completeness and accuracy of their assignments. This report allows you to quickly identify these errors and view any unassigned values.

Business Units

A business unit is a unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy. When a business function produces financial transactions, a business unit must be assigned a primary ledger, and a default legal entity. Each business unit can post transactions to a single primary ledger, but it can process transactions for many legal entities. Normally, it will have a manager, strategic objectives, a level of autonomy, and responsibility for its profit and loss. You define business units as separate task generally done after the accounting setups steps.

The business unit model:

- Allows for flexible implementation
- Provides a consistent entity for controlling and reporting on transactions
- Enables sharing of sets of reference data across applications

For example, if your company requires business unit managers to be responsible for managing all aspects of their part of the business, then consider using two balancing segments, company and business unit to enable the production of business unit level balance sheets and income statements.

Transactions are exclusive to business units. In other words, you can use business unit as a securing mechanism for transactions. For example, if you have an export business that you run differently from your domestic business, use business units to secure members of the export business from seeing the transactions of the domestic business.

Creating Primary Ledgers: Example

Create a primary ledger as your main record-keeping ledger. Like any other ledger, a primary ledger records transactional balances by using a chart of accounts with a calendar, currency, and accounting rules implemented in an accounting method. The primary ledger is closely associated with the subledger transactions and provides context and accounting for them.



Scenario

Your company, InFusion Corporation is implementing Oracle Fusion Applications. You have been assigned the task of creating a primary ledger for your InFusion America entity.

- 1. Navigator > Define Accounting Configurations > Manage Primary Ledgers > Go to Task.
- 2. Click the Create icon.
- 3. Enter the following values:

Field	Value
Name	InFusion America
Description	InFusion America primary ledger for recording transactions.
Chart of Accounts	InFusion America Chart of Accounts
Accounting Calendar	Standard Monthly
Currency	USD
Accounting Method	Standard Accrual

4. Click Save and Edit Task List to navigate back to the accounting configuration task list.

Note: You cannot change the chart of accounts, accounting calendar, or currency for your ledger after you save your ledger.

Specify Ledger Options

Specifying Ledger Options: Worked Example

This example demonstrates specifying the ledger options for your primary ledger. Your company, InFusion Corporation, is a multinational conglomerate that operates in the United States (US) and the United Kingdom (UK). InFusion has purchased an Oracle Fusion Enterprise Resource Planning (ERP) solution including Oracle Fusion General Ledger and all of the Oracle Fusion subledgers.

After completing your InFusion America Primary Ledger, select **Specify Ledger Options** under the Define Accounting Configuration task list on the Functional Setup Manager page.

Note: Both primary and secondary ledgers are created in the same way and use the same user interface to enable their specific ledger options.



Reviewing General Region Options

- 1. Accept the Name and Description defaults for the ledger selected.
- 2. Review the Currency and Chart of Accounts for the specified ledger, which are automatically populated.

Setting Accounting Calendar Region Options

- 1. Review the **Accounting Calendar** that defaults from your ledger.
- 2. Select Jan-2011 as the First Open Period for your ledger.

Select a period after the first defined period in the ledger calendar to enable running translation. You cannot run translation in the first defined period of a ledger calendar. In this example, your calendar began with Jan-2010.

3. Enter 3 for the Number of Future Enterable Periods.

Any value between 0 and 999 periods can be specified to permit entering journals but not posting them in future periods. Minimize the number of open and future periods to prevent entry in the wrong period.

Selecting the Subledger Accounting Region Options

- 1. Accept the default **Accounting Method** from your ledger.
- 2. Select US American English as your Journal Language.

Completing the Period Close Region Options

1. Enter your Retained Earnings Account: 101-00-31330000-0000-0000-0000.

This account is required for general ledger to move the revenue and expense account balances to this account at the end of the accounting year.

2. Enter your Cumulative Translation Adjustment Account: 101-00-31350000-0000-0000-0000.

The Cumulative Translation Adjustment (CTA) account is required for ledgers running translation.

3. Do not enter a **Default Period End Rate Type** or **Default Period Average Rate Type**.

The values entered here are used as the default for balance level reporting currency processing. InFusion America Primary Ledger is using the subledger level reporting currency processing.

Specifying the Journal Processing Region Options

1. Specify the Balance options as outlined in the following table.

Option	Setting
Enable Suspense	General Ledger
Default Suspense Account	101-00-98199999-0000-0000-0000
Rounding Account	101-10-98189999-0000-0000-0000
Entered Currency Balancing Account	101-10-98179999-0000-0000-0000
Balancing Threshold Percent	10

2. Click all the following Entry options listed in the table.



Option	Description	
Enable journal approval	Click to enable journal approval functionality. Approval rules must be created in the Oracle Fusion Approvals Management (AMX).	
Notify when prior period journal	Notify the user when a prior period date is selected on a journal entry.	
Allow mixed and statistical journals	Enter both monetary and statistical amounts on the same line in a journal entry.	
Validate reference date	Requires a reference date in an open or future enterable period.	

- 3. Click the **Separate journals by accounting date during journal import** for the Import option to create individual journal entries for each accounting date.
- 4. For the Reversal options, select InFusion America Accrual Set from the list of values in the Journal Reversal Criteria Set field and click the Launch AutoReverse after open period to reverse accrual journal entries automatically when a new period is opened.
- 5. Click the **Enable intercompany accounting** for the Intercompany option to enable automatic balancing for primary, second, and third balancing segments) on intercompany journals and transactions.

To complete the intercompany accounting functionality, you must define intercompany rules.

Related Topics

- Journal Posting Process: How Single Currency Journals Are Balanced
- Journal Posting Process: How Multicurrency Journals Are Balanced

FAQs for Specify Ledger Options

What happens if I change the cumulative adjustment account?

Changing the cumulative translation adjustment (CTA) account is a very significant revision to your accounting configuration and should be avoided if possible. To prevent data corruption, your CTA can only be changed if you delete translated balances.

Related Topics

Deleting Translated Balances: How It Works



What happens if I change the retained earnings account?

Changing the retained earnings account is a very significant revision to your accounting configuration and should be avoided if possible. To prevent data corruption, your retained earnings account can only be changed if you first perform the following set of steps:

- 1. To reverse what was closed out to the incorrect retained earnings account, enter and post journals to bring the ending balances for your income statement accounts to zero at the end of each accounting year. Use a temporary account, such as a suspense account, for the offsetting amount.
- 2. Update the retained earnings account.
- 3. Reverse the journal entries used to bring the income statement accounts' ending account balances to zero to reinstate each accounting year's profit and loss, and now close these out to the new retained earning's account in the following accounting year.
 - Note: The recommended reversal method for the journals is Change Sign.
- **4.** If you have translated balances for the ledger, follow the process of deleting translated balances for a change in the retained earnings account.

Related Topics

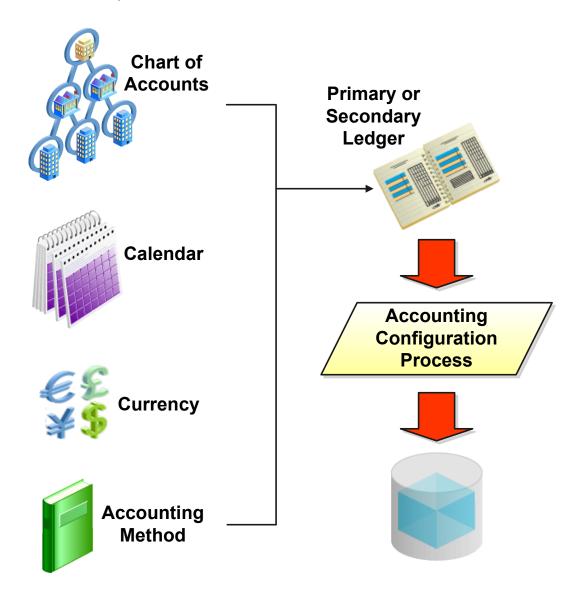
• Deleting Translated Balances: How It Works

Review and Submit Accounting Configuration



Oracle Fusion General Ledger Balances Cube: Overview

A balances cube is an online analytical application (OLAP) database that maintains financial balances in a multidimensional database. A balances cube is created when an accounting configuration is submitted for a primary or secondary ledger that uses a new unique combination of chart of accounts and calendar.



The balances cubes are named after the chart of accounts they contain.

A balances cube:

- Stores your financial balances in a multidimensional enable real time, interactive financial reporting and analysis.
- Preaggregates your balances at every possible point of summarization. Preaggregation ensures immediate access to financial data and eliminates the need for an external data warehouse for financial reporting.



- Is uniquely identified by the combination of a chart of accounts and an accounting calendar. Average balances are tracked in a separate balances cube.
- Is automatically synchronized by the following general ledger processes: posting, open period, and translation.
- Consists of a set of defining business entities called dimensions. Dimensions in a cube determine how data is accumulated for reporting and analytical purposes
- Are referred to as an application or database connection in the user interfaces for:
 - Financial Reports
 - Smart View
 - Calculation Manager
- Note: Account balances were maintained in relational tables in Oracle E-Business Suite General Ledger. The Oracle Fusion General Ledger updates balances in real time to the relational tables and the General Ledger Balances cubes. Most inquiry and reporting are now performed from the General Ledger Balances cubes and not from the relational tables.

Standard Balances Cube Dimensions: Explained

A balances cube consists of a set of defining business entities called dimensions. This table details the dimensions that are available for creating financial reports, Smart View queries, and allocations using multidimensional cubes.

Dimension	Description	Example
Accounting Period	Based upon the calendar of the ledger or ledger set. Report on years, quarters, or periods.	2012Quarter-1Jan-12
Ledger or Ledger Set	Used to select a ledger for the reporting. Multiple ledgers may be in the same cube if they share a common chart of accounts.	Vision North America Ledger SetVision US Primary Ledger
Chart of Accounts Segments	Uses a separate dimension for each of the segments from the charts of accounts. Organized by hierarchy. A default hierarchy is provided that includes all detail segment values. Hierarchies published in the Publish Account Hierarchies user interface are included.	Company: Vision America: 101Cost Center: Sales: 400Account: Cash: 1110
Scenario	Indicates if the balances represented are actual or budget amounts. Allocation-related dimensions are predefined members and required for allocation solutions. Allocation dimensions are not used directly by end users.	Budget 2012ActualsForecast 2013
	Budget scenario dimension members are user-defined in the Oracle Fusion Applications value set called Accounting Scenario. The dimension members appear in	



Dimension	Description	Example
	the cube after running the Create Scenario Dimension Members process.	
Balance Amount	Indicates if the value is the beginning balance, period activity, or ending balance. Debit, Credit, and Net amounts are available for reporting.	 Beginning Balance (DR, CR, or Net) Period Activity (DR, CR, or Net) Ending Balance (DR, CR, or Net)
Amount Type	Indicates whether the amounts represent Base, Period to Date, Quarter to Date, or Year to Date.	BasePTD: Period to DateQTD: Quarter to DateYTD: Year to Date
Currency	Used to select the wanted currency for the balances.	All ISO CurrenciesUSD: US DollarJPY: Japanese Yen
Currency Type	Used to select the currency type of the balances.	TotalEnteredConverted From (for each ISO currency)

Note: Dimensions are predefined and new ones cannot be added.

Creating GL Balances Cubes: Examples

There are two types of Oracle Fusion General Ledger Balances cubes: Standard Balances cubes and Average Balances cubes.

Standard Balances Cubes

A new standard balances cube is created whenever an accounting configuration is submitted for a ledger, either primary or secondary, that uses a new unique combination of chart of accounts and accounting calendar. Cubes are named after the chart of accounts.

Two examples of standard balance cubes:

- 1. The chart of accounts, InFusion US Chart of Accounts has a related cube entitled, InFusion US Chart of Accounts. If a chart of accounts is used by multiple ledgers with different calendars, the cube names are distinguished by appending a number to their names.
- 2. The InFusion US Chart of Accounts is used by two different ledgers, each of which has a different accounting calendar, one with a standard calendar year ending December 31st and the other with a fiscal year ending May 31st. Two cubes are created. The cubes are named InFusion US Chart of Accounts and InFusion US Chart of Accounts 2.

Average Balances Cubes

Average balances cubes use different dimensions than the standard balances cubes therefore require their results be stored in separate cubes. If the average balances option is enabled for a ledger, a second average balances cube is automatically created based upon the same criteria of a unique combination of chart of accounts and accounting calendar. Average balances cubes are named with ADB (average daily balances) plus the name of the associated chart of accounts.



For example, for a chart of accounts, InFusion US Chart of Accounts, the average balances cube name is **ADB InFusion US Chart of Accounts**. Numbers are appended to the name when more than one calendar is used by the same chart of accounts. The numbering is determined and shared with the related standard balances cubes. The standard cube called **InFusion US Chart of Accounts 3** has a corresponding average balance cube entitled **ADB InFusion US Chart of Accounts 3**.

Customer Specific GL Balances Cube Dimension Values: Points to Consider

In creating your cube, take in to consideration the dimensions that are you, the customer, define and those that are partially or completely predefined by Oracle Fusion Applications.

- Two dimensions, Chart of Accounts and Scenario, have customer specific dimension values and require user procedures to populate the cube
- Accounting Period, Currency, and Ledger dimensions are customer specific, but are automatically created in the cubes from the changes made in the applicable user interfaces.
- Other dimensions, such as Amount Type and Balance Amount, have only predefined members.

The following are points to consider in setting up and populating cubes for the Chart of Accounts and Scenario dimension members.

Create Account Hierarchies

- Account hierarchies for your chart of account dimensions are created in the Manage Account Hierarchies page or the Manage Trees and Trees Versions page after setting up your value sets, chart of accounts, and values.
- Create account hierarchies (trees) to identify managerial, legal, or geographical relationships between your value set values.
- Define date-effective tree versions to reflect organizational changes within each hierarchy over time.
- The tree version must be in an active or inactive status to be published. Draft versions cannot be published.
- Note: From your implementation project, Navigate > Define Common Applications Configuration > Define Enterprise Structures > Define Financial Reporting Structures > Define Chart of Accounts > Manage Account Hierarchies > Go to Task.

Publish Account Hierarchies

Next, after defining the tree versions, publish account hierarchies (tree versions) to the cube. Before publishing, set the following profile option: **GL: Set Cube Alias to Segment Value Description**. Consider the following points when setting the profile option.

- The member alias displayed in the Oracle Fusion General Ledger balances cubes.
 - Yes displays only the segment value description, for example: Cash.
 - No displays value plus the segment value description, for example: 1110 Cash.
- When you change the profile option setting, republish the hierarchy.
- Default value is No.



- **No** is necessary when there are duplicate segment value descriptions in the same value set. The descriptions are stored in the **Alias** in the cube. Aliases must be unique in a dimension in the cube. For example, if the description West existed for more than two segment values in the **Cost Center** value set the entire cube creation process fails.
 - Note: Essbase cubes display only the first 80 characters in the description, so the first 80 characters in the description must be unique.
- If duplicate segment value descriptions exist, the cube can be created with the profile set to No. At anytime the value of the profile can be changed to Yes after the duplicates are removed. At that time, the account hierarchies (tree versions) must be published or republished to the cube.
- If you set the profile value to Yes, and then decide to rebuild a cube from a process run in Enterprise Scheduler or create a cube by submitting an accounting configuration and duplicate descriptions exist, the create cubes process errors. You have two options at that point:
 - Set the profile value to No and run the Create Cubes process. For a new ledger with a new chart of accounts
 and accounting calendar combination, submit the accounting configuration.
 - Remove the duplicates across all dimensions, not just a single segment. Then run the Create Cubes or resubmit the accounting configuration.

From your implementation project, Navigate > Define Common Applications Configuration > Define Enterprise Structures > Define Financial Reporting Structures > Define Chart of Accounts > Publish Account Hierarchies > Go to Task. Use the Publish Account Hierarchies page to search and publish account hierarchies.

- Select the **Publish** check box. This is indicator of what to include in balances cube by selecting the check box or what you don't want to include in balances cube by removing the check from the check box.
- Select the rows. Functionality allows for select multiple rows to be selected.
- Select the **Publish** button to update the balances cubes. A process is generated.
- Navigator > Tools > Scheduled Processes to monitor the process.
- Note: Use Smart View to verify that the account hierarchies were correctly published.

Define Scenario Dimension Members

The customer specific Scenario dimension members are defined in the **Manage Value Sets** and **Manage Values** pages in the value set called **Accounting Scenario.** Any customer specific Scenario dimension is included in all balances cubes.

Best practice is to setup the customer specific Scenario members in this value set before you create your first ledger and run the **Accounting Configuration** process. The **Accounting Configuration** process generates your balances cubes.

If the cubes already existed, you can run the **Create Scenario Dimension Members** process to update the balances cubes for the new members.

Note: If you are on a release before the **Create Scenario Dimension Members** process is available, you have to rebuild the cubes with the **Create Cubes** process to add the Scenario dimension in the cube. Follow the guidelines for creating cubes before running process.



Default Dimension Values in Reporting: Explained

For Standard Balances Cube dimensions, there are default values that if used in Smart View and on financial reports created in Financial Reporting Studio cause #MISSING to appear when balances are returned on a report output. If #MISSING appears, check that all dimensions are properly set. Particularly, check the Accounting Period, Ledger, Scenario, and Currency dimensions, which all must have a value selected other than the default top level value called Gen1.

The following table lists the available and default dimension values as well as providing guidance on selecting the correct dimensions.

Dimension	Possible Values	Default Value	Additional Information
Accounting Period	Years, quarters, and period	Accounting Period = Accounting Period (Gen1)	You must always select an accounting period for each financial report including User Point of View, Smart View query, or allocation including Point of View. If you do not specify a valid Accounting Period, the financial reports, Smart View queries, and Account Inspector displays #MISSING.
Ledger	Includes ledgers and ledger sets	Ledger = Ledger (Gen1)	If you do not specify a specific Ledger or Ledger Set, the financial reports, Smart View queries, and Account Inspector queries display #MISSING
Chart of accounts dimensions		Highest level (Gen1)	There is a separate dimension for each segment of the chart of accounts used by the cube. Each segment has a default account hierarchy that includes all the detail values for the segment but not parent values. Only account hierarchies (tree versions) published from the Publish Account Hierarchies user interface are available in the cube.
Scenario		Scenario = Scenario represents the sum of all values: Actual + Allocated + Total for Allocations + Budget + Forecast. Select a value for this dimension.	Seeded values are Actual, Allocated, and Total for Allocated. Additional scenario values for Budget, Forecast Q1, and Forecast Q2 may be available if they have been added to the Accounting Scenario value set. These additional values will
Balance Amount	 Beginning Balance DR, CR, or Net Period Activity DR, CR, or 	Scenario dimension. Balance Amount = Balance Amount (Gen1) which is the equivalent of Ending Balance.	be published to every cube.



Dimension	Possible Values • Ending Balance DR, CR, or Net • Balance Amount (same as Ending Balance)	Default Value	Additional Information
Amount Type	Base, PTD, QTD, YTD	Amount Type = Amount Type (Gen1) which is the equivalent of Base.	Base is necessary because this is the value used to store from posting all balances at the lowest level. PTD, QTD, and YTD are calculated values.
Currency	All ISO currencies (250+).	Highest level (Gen1)	Similar to Accounting Period and Ledger, there may not be an appropriate default for Currency since different Ledgers have different ledger currencies. If you don't specify a valid currency in a financial report, Smart View query, or allocation, you will get a result = #MISSING.
Currency Type	Total, Entered, and Converted from for each ISO currency = each ISO currency.	Currency Type = Currency Type (Gen1), which is the equivalent of Total.	

Average Balances Cubes Dimensions

The following dimensions are included in the average balances cube in this order. Most comments from standard cube are applicable below unless stated.

- Accounting Period: Valid values are years, quarters, accounting period and day, which is equivalent to accounting date.
- Ledger
- Separate dimension for each Chart of accounts segment
- Scenario
- Amount Type valid values are: PATD, QATD, and YATD
- Currency
- Currency Type

GL Balances Cube Terminology: Explained

Levels and generations are cube terminology used to describe hierarchies in Oracle Fusion General Ledger balances cubes.

Levels

Levels are used to describe hierarchy levels. Levels are numbered from the lowest hierarchy level. For example, the detail chart of accounts segment values are Level 0. The immediate parent is Level 1; the next parent is Level 2.



Generations

Generations (Gen) describe hierarchy levels starting with the top of the hierarchy and moving down through the generations of the same.

An example for generations is as follows:

- Accounting Period = Accounting Period is Gen 0
 - Year is Gen 1
 - Quarter is Gen 2
 - Period is Gen 3
- Ledger = Ledger is Gen 1
 - o All Ledgers is Gen 2
 - VF USA is Gen 3
 - Ledger Set (any ledger set) is Gen 2
- Company = Company is Gen 1
 - o All Company Values, for example detail values, is Gen 2
 - 11010 is Gen 3
 - o Tree 1 V1 is Gen 2
 - [Tree 1 V1].[10000] is Gen 3
 - [Tree 1 V1].[10000].[11000] is Gen 4
 - Tree 2 V1 is Gen 2
 - [Tree 2 V1].[10000] is Gen 3
 - [Tree 1 V1].[10000].[11000] is Gen 4
- Note: Similar levels and generations apply to the other dimension, including chart of accounts dimensions and those that are not chart of accounts dimensions.

Using Dimension Values in Reporting: Examples

By using combinations of values for the Accounting Period, Balance Amount, and Amount Type dimensions, you can derive different amounts to meet financial reporting requirements. In some cases, more than one combination of query values can



return the wanted information. Some duplication in the calculations for the balances cube exists to ensure complete reporting requirements.



Reporting needs can be completed using the Balance Amount dimension, Amount Type equal to Base, and the Accounting Period equal to Year, Quarter, or Month.

However, the Amount Type dimension is still required for the following reasons:

- To get the complete QTD (Quarter to Date) reporting requirements for the first and second month in a quarter. Complete QTD can only be achieved using QTD and Accounting Period = Month.
- PTD (Period to Date) and YTD (Year to Date) are also available to ensure more consistency reporting with the E-Business Suite Release 12.

Obtaining Quarter Information

This example shows how to obtain quarter information.

- QTD for the first and second months of a quarter can only be achieved using Amount Type dimension equal to QTD.
- The end of quarter information can be derived from the Accounting Period dimension equal to Quarter or with the Amount Type equal QTD.

Obtaining End of Year Information

This example shows how to obtain end of year information.

- Accounting Period equal to Year and Balance Amount equal to Ending Balance and Amount Type equal to Base or YTD.
- Accounting Period equal to a specific month and Balance Amount equal to Period Activity and Amount Type equal to YTD.
- When the Balance Amount equals the Balance Amount or the Balance Amount equals the Ending Balance, this results in an Ending Balance.
- Note: Ending Balance is always the ending balance regardless of Amount Type member setting or Accounting Period member setting (period, quarter or year).

Standard Balances: Example

You have loaded your Oracle E-Business Suite Release 12 balances into your new Oracle Fusion Standard Balances Cube. Now you want to query the results to verify that the data was loaded correctly.



Scenario

The tables show the amounts loaded into the balances cube from your R12 balances and the calculated balances for:

- Ending balance.
- Year to date (YTD).
- Quarter to date (QTD.

The first table is the balance sheet-based balances and the second table is for income statement-based balances.

Oracle E-Business Suite R12 Balances

Balance Sheet Balances

Month	Beginning Balance Loaded to Cube	Activity Loaded to Cube	Ending Balance	Calculate YTD	QTD
Jan	200	20	220	20	20
Feb	220	30	250	50	50
Mar	250	25	275	75	75
Apr	275	50	325	125	50
May	325	40	365	165	90
Jun	365	45	410	210	135
Jul	410	100	510	310	100
Aug	510	200	710	510	300
Sep	710	140	850	650	440
Oct	850	150	1000	800	150
Nov	1000	100	1100	900	250
Dec	1100	400	1500	1300	650

Income Statement Balances

Month	Beginning Balance Loaded to Cube	Activity Loaded to Cube	Ending Balance	Calculate YTD	QTD
Jan	0	70	70	70	70



Month	Beginning Balance Loaded to Cube	Activity Loaded to Cube	Ending Balance	Calculate YTD	QTD
Feb	70	20	90	90	90
Mar	90	30	120	120	120
Apr	120	100	220	220	100
May	220	200	420	420	300
Jun	420	250	670	670	550
Jul	670	50	720	720	50
Aug	720	300	1020	1020	350
Sep	1020	130	1150	1150	480
Oct	1150	110	1260	1260	110
Nov	1260	200	1460	1460	310
Dec	1460	500	1960	1960	810

Analysis

The following two tables show the results available from Oracle Fusion General Ledger from your R12 loaded and aggregated balances in the balances cube. The first table is the balance sheet-based balances and the second table is for income statement-based balances.

Oracle Fusion Balances

Balance Sheet Balances

Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
May	Beginning Balance	YTD	200	Jan Beginning Balance	
May	Period Activity	YTD	165	Jan to May Activity	YTD-Actual as of May
May	Ending Balance	YTD	365	May Ending Balance	YTD-Actual as of May
May	Beginning Balance	QTD	275	Apr Beginning Balance	



Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
May	Period Activity	QTD	90	Apr-May Activity	QTD-Actual as of May
May	Ending Balance	QTD	365	May Ending Balance	QTD-Actual as of May
May	Beginning Balance	PTD	325	May Beginning Balance	
May	Period Activity	PTD	40	May Activity	PTD-Actual as of May
May	Ending Balance	PTD	365	May Ending Balance	PTD-Actual as of May
May	Beginning Balance	Base	325	May Beginning Balance	
May	Period Activity	Base	40	May Activity	PTD-Actual as of May
May	Ending Balance	Base	365	May Ending Balance	YTD-Actual as of May
Q2	Beginning Balance	QTD	275	April Beginning Balance	
Q2	Period Activity	QTD	135	Always Apr-Jun	QTD-Actual as of Jun
Q2	Ending Balance	QTD	410	Ending Balance always as of Jun	QTD-Actual as of Jun
Q2	Ending Balance	YTD	410	Ending Balance always as of Jun	YTD-Actual as of Jun
Q2	Beginning Balance	Base	275	Apr Beginning Balance	
Q2	Period Activity	Base	135	Always Apr-Jun	QTD-Actual as of Jun
Q2	Ending Balance	Base	410	Ending Balance always as of Jun	YTD-Actual as of Jun
Year	Beginning Balance	YTD	200	Jan Beginning Balance	



Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
Year	Period Activity	YTD	1300	Jan-Dec Activity	YTD-Actual as of Dec
Year	Ending Balance	YTD	1500	Ending Balance always as of Dec	YTD-Actual as of Dec
Year	Beginning Balance	Base	200	Jan Beginning Balance	
Year	Period Activity	Base	1300	Jan-Dec Activity	YTD-Actual as of Dec
Year	Ending Balance	Base	1500	Ending Balance always as of Dec	YTD-Actual as of Dec

Income Statement Balances

Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
May	Beginning Balance	YTD	0	Jan Beginning Balance	
May	Period Activity	YTD	420	Jan to May Activity	YTD-Actual as of May
May	Ending Balance	YTD	420	May Ending Balance	YTD-Actual as of May
May	Beginning Balance	QTD	120	Apr Beginning Balance	
May	Period Activity	QTD	300	Apr-May Activity	QTD-Actual as of May
May	Ending Balance	QTD	420	May Ending Balance	QTD-Actual as of May
May	Beginning Balance	PTD	220	May Beginning Balance	
May	Period Activity	PTD	200	May Activity	PTD-Actual as of May
May	Ending Balance	PTD	420	May Ending Balance	PTD-Actual as of May



Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
May	Beginning Balance	Base	220	May Beginning Balance	
May	Period Activity	Base	200	May Activity	PTD-Actual as of May
May	Ending Balance	Base	420	May Ending Balance	YTD-Actual as of May
Q2	Beginning Balance	QTD	120	Apr Beginning Balance	
Q2	Period Activity	QTD	550	Period Activity Always Apr-Jun	QTD-Actual as of Jun
Q2	Ending Balance	QTD	670	Ending Balance always as of Jun	QTD-Actual as of Jun
Q2	Ending Balance	YTD	670	Ending Balance always as of Jun	YTD-Actual as of Jun
Q2	Beginning Balance	Base	120	Apr Beginning Balance	
Q2	Period Activity	Base	550	Period Activity always Apr-Jun	QTD-Actual as of Jun
Q2	Ending Balance	Base	670	Ending Balance always as of Jun	YTD-Actual as of Jun
Year	Beginning Balance	YTD	0	Jan Beginning Balance (always zero for Income Statement)	
Year	Period Activity	YTD	1960	Jan-Dec Activity	YTD-Actual as of Dec
Year	Ending Balance	YTD	1960	Ending Balance always as of Dec	YTD-Actual as of Dec
Year	Beginning Balance	Base	0	Jan Beginning Balance (always zero for Income Statement)	
Year	Period Activity	Base	1960	Jan-Dec Activity	YTD-Actual as of Dec



Accounting Period	Balance Amount	Amount Type	Value	Comments	R12 Amount Type Equivalent
Year	Ending Balance	Base	1960	Ending Balance always as of Dec	YTD-Actual as of Dec

Invalid Balance Cube Dimension Combinations: Explained

By using various combinations for the Accounting Period, Balance Amount, and Amount Type dimensions, you can derive different amounts to meet financial reporting requirements.

Balances cubes do not return data for these combinations:

Accounting Period	Balance Amount	Amount Type
Year	Beginning Balance	QTD
Year	Period Activity	QTD
Year	Ending Balance	QTD
Year	Beginning Balance	PTD
Year	Period Activity	PTD
Year	Ending Balance	PTD
Q2	Beginning Balance	YTD
Q2	Period Activity	YTD
Q2	Beginning Balance	PTD
Q2	Period Activity	PTD
Q2	Ending Balance	PTD

GL Balance Cube Related Processes Run from User Interfaces

This list describes the processes run from the application pages (user interfaces) used to create, rebuild, publish, and maintain Oracle Fusion General Ledger balances cubes, including automatically run child processes.

Create Cube from Accounting Configuration Page



Navigate > Setup and Maintenance work area > Define Common Applications Configuration > Define Ledgers > Define Accounting Configuration > Review and Submit Accounting Configuration

Parent Process	Child Process	Description	Cube Type
Create Accounting Configuration		Updates the underlying table and permanently saves the settings after the accounting configuration has been completed. Some settings cannot be changed or deleted.	Standard
	Create Accounting Configuration: Create Cubes	Creates the accounting configuration and a balances cube when setup is completed.	Standard and ADB, if enable, in same request

Publish Chart of Accounts from Publish Account Hierarchies Page

Navigate > Setup and Maintenance work area > Define Common Applications Configuration > Define Enterprise Structures > Define Financial Reporting Structures > Define Chart of Counts > Publish Account Hierarchies

Parent Process	Child Process	Description	Cube Type
Publish Chart of Account Hierarchies		Creates the structural hierarchy for the chart of accounts.	Standard and ADB, if enable, in same request
	Publish Chart of Account Hierarchy Versions	One runs for each combination of value set and chart of accounts.	Standard and ADB, if enable, in same request

Create Ledger Set Processes from Ledger Set Page

Navigate > Setup and Maintenance work area > Define Common Applications Configuration > Define General Ledger Options > Manage Ledger Sets

Parent Process	Child Process	Description	Cube Type
Create Ledger Set		Updates the underlying tables and creates the ledger set.	Standard and ADB, if enable, in same request
	Create Accounting Configuration: Process Accounting Configuration	Runs processes necessary to process accounting configuration and ledger set data.	Not applicable to the Cube
	Create Ledger Set Dimension Members	Creates and updates ledger set members in a balances cube.	Standard and ADB, if enable, in same request

Other Requests Submitted from Pages or Processes



Parent Process	Child Process	Description	Cube Type	Notes
Create Ledger Dimension Member for Reporting Currency		Creates ledger dimension member for reporting currency.	Standard and ADB, if enable, in same request	Navigate > Setup and Maintenance work area > Define Common Applications Configuration > Define Ledgers > Define Accounting Configuration > Manage Reporting Currencies > Create Reporting Currency page, run for new reporting currencies added after accounting configuration has completed.
Update Ledger Dimension Member for Reporting Currency		Updates ledger dimension member for reporting currency.	Standard and ADB, if enable, in same request	Navigate > Setup and Maintenance work area > Define Common Applications Configuration > Define Ledgers > Define Accounting Configuration > Manage Reporting Currencies->Update Reporting Currency page, this job runs for new reporting currencies added after accounting configuration has completed.
	Publish Chart of Accounts Dimension Members: Detailed Values Only	Publishes chart of accounts dimension member changes to balances cubes and updates dimension members for any new or changed segment values.	Standard and ADB, if enable, in same request	Run only from the journal posting program when detailed values do not exist in the GL balances cube during posting.

GL Balance Cube Related Processes Run in ESS

This list describes the processes used to create, rebuild, publish, and maintain Oracle Fusion General Ledger balances cubes, including automatically run child processes.

Create Cube Processes Run in Oracle Enterprise Scheduler Service (ESS)

Parent Process	Child Process	Description	Cube Type
Create Cube		Creates or rebuilds the balances cube based on the combination of chart of accounts and accounting calendar.	Standard



Parent Process	Child Process	Description	Cube Type
		Automatically runs the child processes below for standard and average daily balance (ADB) cubes, if enabled.	
	Create Cube: Initialize Cube	Starts the process to import data into the balances cube.	Standard
	Create Cube: Create Calendar Dimension Members and Hierarchies	Creates the calendar dimension members and hierarchies for a balances cube.	Standard
	Create Cube: Create Ledger Dimension Members	Creates the ledger dimension members for a balances cube.	Standard
	Create Cube: Publish Chart of Accounts Dimension Members and Hierarchies	Publishes chart of accounts and hierarchy changes to balances cubes and updates dimension members for any new or changed segment values.	Standard
	Create Average Daily Balance Cube	Determines the amount needed to manually adjust general ledger account average balances to reflect the differences between the original and revalued customer open items.	Average Daily Balance (ADB)
	Create Cube: Initialize Average Balances Cube	Begins the process to import average balances into the balances cube.	ADB
	Create Cube: Create Daily Dimension Members and Hierarchies	Creates the daily calendar dimension members and hierarchies for a balances cube.	ADB
	Create Cube: Create Ledger Dimension Members	Creates the ledger dimension members for a balances cube.	ADB
	Create Cube: Publish Chart of Accounts Dimension Members and Hierarchies	Publishes chart of accounts and hierarchy changes to balances cubes and updates dimension members for any new or changed segment values.	ADB
	Create Cube: Transfer General Ledger Balances to Essbase	Transfer balances to balances cubes.	Standard and ADB, if enable, in same request

Publish Cube Processes Run in ESS



Parent Process	Child Process	Description	Cube Type
Publish Chart of Accounts Dimension Members and Hierarchies		Publishes chart of accounts dimension member and hierarchy changes to balances cubes and updates dimension members for any new or changed segment values.	Standard and ADB, if enable, in same request
	Update Chart of Accounts Dimension Members and Hierarchies	Updates chart of accounts dimension member and hierarchy changes to Essbase.	Standard and ADB, if enable, in same request

Transfer Cube Process Run in ESS

Parent Process	Description	Cube Type
Transfer General Ledger Balances to Essbase	Refreshes standard cube (and average balances cube if enabled) in the General Ledger balances cube.	Standard and ADB, if enable, in same request

Other Cube Processes Run in ESS

Parent Process	Descriptions	Cube Type
Create Accounting Period Dimension for Standard Cube	Creates the accounting period dimension members.	Standard
Create Ledger Dimension Members	Creates and updates ledger dimension members including primary ledgers, secondary ledgers, reporting currencies, and ledger sets in the balances cubes.	Standard and ADB, if enable, in same request
Create Currency Dimension Members	Creates and updates all currencies in every balances cube.	Standard and ADB, if enable, in same request
Create Scenario Dimension Members	Creates and updates all scenario dimension members when new scenarios are created or existing scenarios are changed.	Standard and ADB, if enable, in same request
Create Accounting Period Dimension for Average Daily Balances Cube	Creates the accounting period dimension members in the average daily balances cube.	ADB
Create Rules XML File for BI Extender Automation		Standard and ADB, if enable, in same request

To run cube process, use the following steps the Scheduled Processes work area from the Navigator menu.

- 1. Click the **Schedule New Process** button
- 2. Search on the Process Name.
- 3. Enter the desired parameters.



- 4. Enter the desired process options and schedule.
- 5. Click Submit.

FAQs for Review and Submit Accounting Configuration

How can I inquire on translated balances?

You can query on Currency Type equal to Total and Currency equal to Translated Currency.

When do I rebuild the GL balances cubes?

Carefully consider requirements and work with Oracle Support before rebuilding a balances cube. Use the on-demand programs to rebuild dimension members for each dimension and to refresh balances to the cubes rather than rebuilding the cube. When you rebuild a cube, the process rebuilds both the standard and average balances cubes.

To rebuild cubes, run the **Create Cubes** process. Enter values for the following parameters:

- Chart of Accounts
- Accounting Calendar
- Starting Period

How can I refresh balances in the GL balances cubes?

Run the **General Ledger Transfer Balances to Essbase** process. Select your Ledger or Ledger Set and Starting Period as parameters

Assign Access to Ledger

Assigning Legal Entities and Balancing Segments: Examples

Optionally, assign legal entities and balancing segments to your accounting configuration.

Assign Legal Entities

Assign one or more legal entities to your configuration by following these steps:

- Navigator > Setup and Maintenance work area > Define Ledgers > Define Accounting Configurations >
 Assign Legal Entities task.
- 2. If scope is:
 - Not set: Select Scope link > Assign Legal Entities radio button > In the Primary Ledger drop down Select and Add > Apply and Go To Task > Select your ledger > Save and Close.



- Set, click Go to Task
- 3. Click the Select and Add icon.
- **4.** Enter your legal entity.
- 5. Apply > Done.
- 6. Save and Close.

Assign Balancing Segments to Legal Entities

Assign balancing segment values to your legal entities by following these steps:

- Navigator > Setup and Maintenance work area > Define Ledgers > Define Accounting Configurations >
 Assign Balancing Segment Values to Legal Entities task.
- 2. If scope is:
 - Not set: Select Scope link > Assign Balancing Segment Values to Legal Entities radio button > In the Primary Ledger drop down Select and Add > Apply and Go To Task > Select your ledger > Save and Close.
 - Set, click Go to Task.
- 3. Click the Create icon.
- 4. Select the balancing segment value. Optionally, add a Start Date.
- **5. Save and Close** to close the create page.
- 6. Save and Close.

Assign Balancing Segments to Ledgers

Assign balancing segment values directly to your ledger by following these steps:

- Navigator > Setup and Maintenance work area > Define Ledgers > Define Accounting Configurations >
 Assign Balancing Segment Value to Ledger task.
- 2. If scope is:
 - Not set: Select Scope link > Assign Balancing Segment Value to Ledger radio button > In the Primary Ledger drop down Select and Add > Apply and Go To Task > Select your ledger > Save and Close.
 - Set, click Go to Task.
- 3. Select the balancing segment value.
- 4. Optionally enter a start date.
- 5. Save and Close.
- Note: The balancing segment values that are assigned to the ledger represent nonlegal entity transactions, such as adjustments. If you use legal entities, you must assign balancing segment values to all legal entities before assigning values to the ledger. The only available balancing segment values that can be assigned to ledgers are those not assigned to legal entities.

Related Topics

Balancing Segments: Explained

Manage General Ledger Security



Implementing Financials Security: Overview

Oracle Financials Cloud predefines common job roles such as **Accounts Payable Manager** and **General Accounting Manager**. You can use these roles, modify them, or create job roles as needed. A user can be assigned more than one role, so don't define a role that includes all the accesses needed for every user.

For a listing of the predefined job roles in Oracle Financials Cloud and their intended purposes, see the Oracle Financials Cloud Security Reference Manual in the Oracle Help Center (http://docs.oracle.com).

For more information on securing your applications, see the Oracle ERP Cloud Securing Oracle ERP Cloud guide in the Oracle Help Center (http://docs.oracle.com).

Other Financials Security Considerations: Explained

Common functionality that is not job specific, such as creating expense reports and purchase requisitions, are granted to abstract roles like **Employee**, **Line Manager**, and **Purchase Requestor**.

Oracle Financials Cloud includes the following roles that are designed for initial implementation and the ongoing management of setup and reference data:

- **Application Implementation Manager:** Used to manage implementation projects and assign implementation tasks.
- Application Implementation Consultant: Used to access all setup tasks.
- Note: For the ongoing management of setup and reference data, the Financial Application Administrator, a predefined administrator role, provides access to all financial setup tasks.

Segregation of Duties Considerations

Segregation of duties (SOD) separates activities such as approving, recording, processing, and reconciling results so you can more easily prevent or detect unintentional errors and willful fraud.

Oracle Financials Cloud includes roles that have been defined with a knowledge of a set of SOD policies that are included in the Oracle Cloud Access Controls Governor product. The job roles are based on those commonly defined in business and the duty definitions are defined using the Oracle Cloud SOD policies.

For example, the privilege **Create Payments** is incompatible with the privilege **Approve Invoice**. The predefined **Accounts Payable Manager** role has the privileges of **Force Approve Invoices** and **Create Payments**. When you assess and balance the cost of duty segregation against reduction of risk, you may determine that the **Accounts Payable Manager** role is not allowed to perform force approve invoices and remove this privilege.

To learn more about the policies and roles, see the Oracle Financials Cloud Security Reference Manual in the Oracle Help Center (http://docs.oracle.com).

Data Security Considerations

- Use segment value security rules to restrict access to transactions, journal entries, and balances based on certain values in the chart of accounts, such as specific companies and cost center values, to individual roles.
- Use data access set security for Oracle Fusion General Ledger users to control read or write access to entire ledgers or portions of the ledger represented as primary balancing segment values, such as specific legal entities or companies.



For more information on securing your applications, see the Oracle ERP Cloud Securing Oracle ERP Cloud guide in the Oracle Help Center (http://docs.oracle.com).

General Ledger Security: Explained

General ledger functions and data are secured through job roles, data access sets, and segment value security rules.

Functional Security

Functional security, which is what you can do, is managed using job roles. The following job roles are predefined for Oracle Fusion General Ledger:

- General Accounting Manager
- General Accountant
- Financial Analyst

Each job role includes direct privilege grants, as well as duty role assignments, to provide access to application functions that correspond to their responsibilities. For example, the General Accounting Manager role grants comprehensive access to all General Ledger functions to the general accounting manager, controller, and chief financial officer in your organization.

Data Security

Data security, which controls what action can be taken against which data, is managed using:

- Data access sets
- Segment value security rules

Data access sets can be defined to grant access to a ledger, ledger set, or specific primary balancing segment values associated with a ledger. You decide whether each data access set provides read-only access or read and write access to the ledger, ledger set, or specific primary balancing segment values, which typically represent your legal entities that belong to that ledger. Primary balancing segment values without a specific legal entity association can also be directly assigned to the ledger.

Segment value security rules control access to data that is tagged with the value set values associated with any segment in your chart of accounts.

Security Assignment

Use the Security Console to assign users roles (job roles, as well as roles created for segment value security rules or others). Use the Manage Data Access Set Data Access for Users task to assign users data access sets as the security context paired with their General Ledger job role assignments.

For more information about security assignments, see the Securing Oracle ERP Cloud guide.

Data Access Set Security: Overview

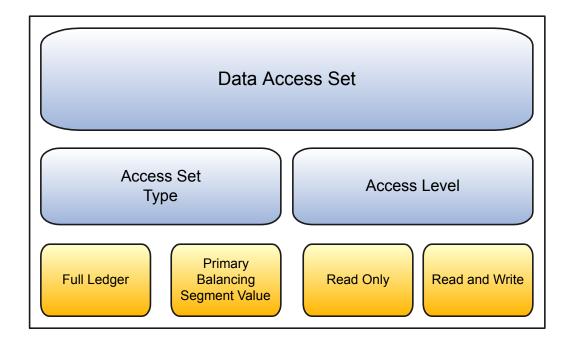
Data Access Sets secure access to ledgers, ledger sets, and portions of ledgers using primary balancing segment values. If you have primary balancing segment values assigned to a legal entity, then you can use this feature to secure access to specific legal entities.



You can combine ledger and ledger set assignments in single data access sets if the ledgers share a common chart of accounts and calendar. If you have primary balancing segment values assigned to a legal entity within the ledger, then you can use data access sets to secure access to specific legal entities. You can also secure access to primary balancing segments assigned directly to the ledger.

When a ledger or ledger set is created, a data access set for that ledger or ledger set is automatically created, giving full read and write access to those ledgers. You can also manually create data access sets to give read and write access, or read-only access to entire ledgers or portions of the ledger represented as primary balancing segment values.

This figure shows how data access sets consist of an access set type and an access level.



The **Full Ledger** access set type provides access to the entire ledger or ledger set. This could be for read-only access or both read and write access to the entire ledger.

The **Primary Balancing Segment Value** access set type provides access to one or more primary balancing segment values for that ledger. This access set type security can be specified by parent or detail primary balancing segment values. The specified parent value and all its descendants, including middle level parents and detail values are secured. You can specify read only, read and write access, or combination of both, for different primary balancing segment values for different ledgers and ledger sets.

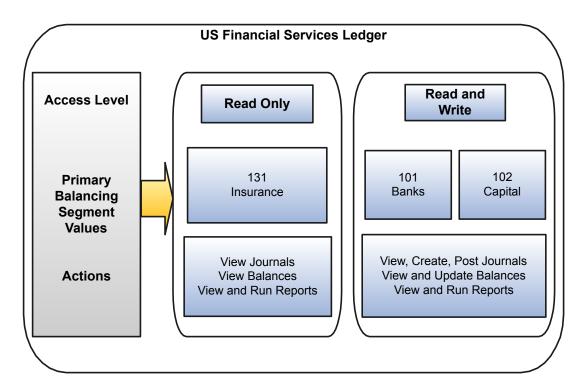
Data Access Set Security: Examples

This example shows a data access set that secures access by using primary balancing segment values that correspond to legal entities.



Scenario

This figure shows a data access set for the US Financial Services Ledger. The access set type is Primary Balancing Segment Value, with each primary balancing segment values representing different legal entities. One access set assignment provides read-only access and the other, read and write access to the corresponding legal entities' primary balancing segment value.



Read-only access has been assigned to primary balancing segment value 131, which represents the Insurance legal entity. Read and write access has been assigned to the other two primary balancing segment values 101 and 102, which represent the Banks and Capital legal entities.

For this data access set, the user can:

- View the journals, balances, and reports for primary balancing segment value 131 for the Insurance legal entity.
- Create journals and update balances, as well as view journals, balances and reports for primary balancing segment value 101 and 102 for legal entities Banks and Capital.
- Note: In financial reporting, the list of ledgers isn't secured by data access sets when viewing a report in Preview mode. Users can view the names of ledgers they don't have privileges to view. However, the data from a secured ledger doesn't appear on the report.

Segment Value Security: Explained

Set up segment value security rules on value sets to control access to parent or detail segment values for chart of accounts segments, also called flexfield segments. Segment value security rules restrict data entry, online inquiry, and reporting.



Secured Value Sets

When you enable security on a value set, access to all values for that value set is denied. To control access to value set values, you enable security on the value set, create conditions, and then assign the conditions to roles. The roles should be created solely for the purpose of segment value security. The roles are then assigned to users.

If a value set is secured, every usage of that value set in a chart of accounts structure instance is secured. For example the same security applies if that value set is:

- Used for two or more segments in the same chart of accounts, such as the primary balancing and intercompany segments
- Shared across different segments of different charts of accounts

Secured Segment Values

Segment value security applies mainly when data is created or updated, and when account combinations are queried. When you have access to secured account values, you can view and use those secured values across all modules of the applications where there are references to accounting flexfields including:

- Transaction entry pages
- Balances and transactions inquiry pages
- Setup pages
- Reports

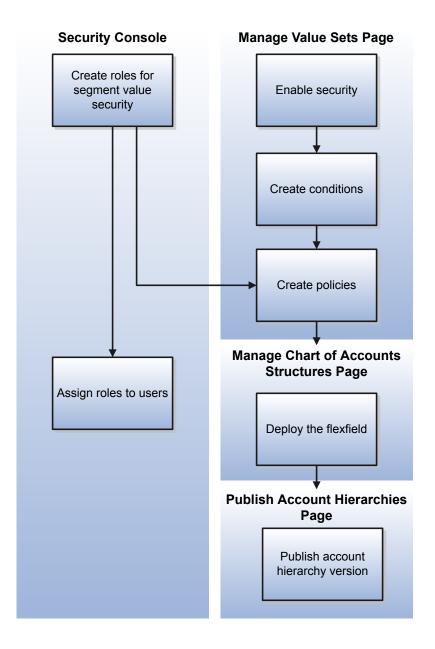
On setup pages, you can still view referenced account combinations with secured account values, even if you haven't been granted access to those secured values. However, if you try to update such references, you can't use those secured values. On reports, you can view balances for secured account values only if you have access to those secured values.

Note: You can enforce segment value security for inquiries and reporting based on any hierarchy, even hierarchies that aren't published to the reporting cube.



Segment Value Security Implementation

This figure shows the steps for defining and implementing security rules for segment values.



To define segment value security roles:

- Create segment value security roles.
- Enable security on the value set.
 - Note: You can enable security only on value sets with a type of Independent.
- Create conditions for the rule.



- Create policies to associate the conditions with the role.
- Deploy the accounting flexfield.
- Publish the account hierarchies.
- Assign the role to users.

Whenever you assign segment value security roles to a user, the rules from the user's assigned roles can be applied together. All of the segment value security roles assigned to a user pertaining to a given value set are simultaneously applied when the user works with that value set. For example, one rule provides access to cost center 110 and another rule provides access to all cost centers. A user with both of these segment value security rules has access to all cost centers when working in a context where that value set matters.

Segment Value Security Conditions

When you create a condition, you specify an operator. The following table describes the operators that you can use.

Operator	Usage
Equal to	Provides access to a specific detail or child value.Don't use to provide access to a parent value.
Not equal to	Provides access to all detail and child values, except the one that you specify.Don't use to provide access to a parent value.
Between	Provides access to a detail range of values.
Is descendant of	 Provides access to the parent value itself and all of its descendants including middle level parents and detail values.
Is last descendant of	Provides access to the last descendants for example, the detail values of a parent value.

Tip: For the operators is descendant of and is last descendant of:

- Specify an account hierarchy (tree) and a tree version to use these operators.
- Understand that the security rule applies across all the tree versions of the specified hierarchy, as well as all hierarchies associated with the same value set of the specified hierarchy.

Segment Value Security: Examples

You can set up segment value security rules on value sets to control access to parent or detail segment values for chart of accounts segments. Segment value security rules restrict data entry, online inquiry, and reporting.

The following example describes why and how you might want to use segment value security.

Securing Values for the Cost Center and Account Segments

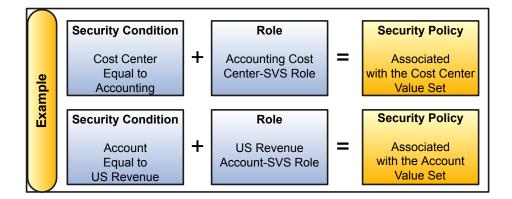
For this scenario, only certain users should have access to the Accounting cost center and the US Revenue account. To create a complete data security policy that restricts segment value access to those users:

1. Plan for the number of roles that represent the unique segment value security profiles for your users. For this scenario, you can create two roles, one for the cost center segment and one for the account segment.



- 2. Use the Security Console to create the roles. Append the text SVS-role to the role names so it's clear the roles are solely for segment value security. For this scenario, you create roles Accounting Cost Center-SVS Role and US Revenue Account-SVS Role.
- 3. Use the Manage Segment Value Security Rules task to enable security on the cost center and account value sets that are associated with the chart of accounts.
- **4.** Create a condition for each value set. For example, the condition for the Accounting cost center is that the cost center is equal to Accounting.
- 5. Create a policy to associate the conditions to the roles. For example, create a policy to assign the condition for the Accounting cost center to the role Accounting Cost Center-SVS Role.
- **6.** Use the Security Console to assign the appropriate role to the appropriate user. For example, assign the role Accounting Cost Center-SVS Role to the users who should have access to the Accounting cost center.

This figure shows how the conditions and roles combine to create the security policies for this scenario.



Enabling Security on a Chart of Accounts: Worked Example

This example demonstrates how to enable security on a chart of accounts to control access to specific segment values.

The following table summarizes the key decisions for this scenario.

Decisions to Consider	In This Example
Which segment in the chart of accounts must be restricted?	Cost center
Which cost center values have to be granted to different users?	 Child values 110 to 120 Child value 310 Parent value 400 and all its children All cost centers
What's the name of the value set for the Cost Center segment?	Cost Center Main
What's the name of the user who can access cost centers 110 to 120?	Casey Brown
What's the name of the tree for the accounting flexfield?	All Corporate Cost Centers



Decisions to Consider	In This Example
What version of the tree hierarchy does the condition apply to?	V5

Summary of the Tasks and Prerequisites

This example includes details of the following tasks you perform when defining and implementing segment value security.

- 1. Define roles for segment value security rules.
- 2. Enable segment value security for the value set.
- 3. Define the conditions.
- Define the policies.
- 5. Deploy the accounting flexfield.
- 6. Publish the account hierarchies.
- 7. Assign segment value security roles to users.

Perform the following prerequisites before enabling security on a chart of accounts:

- To work with the Security Console, you need the IT Security Manager role assigned to your user setup.
- To work with value sets and profile options, you need the Financial Application Administrator role.
- Set the Enable Data Security Polices and User Membership Edit profile to Yes.

Defining Roles for Segment Value Security Rules

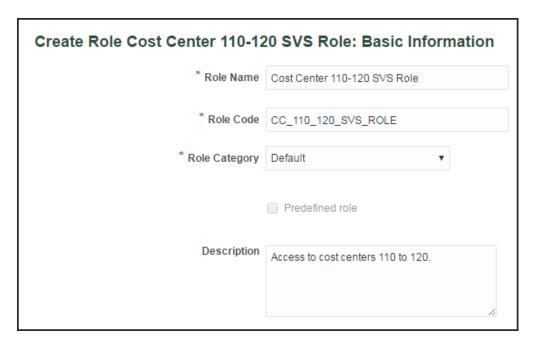
To create a complete data security policy, create the roles first so that they're available for assignment to the segment value security rules.

- 1. In the Tools work area, open the Security Console.
- 2. Perform the following steps four times to create four roles.
- 3. Click Create Role.
- On the Create Role page, complete the fields as shown in this table, and then click Next, Next,
- 5. Click **OK**.

Field	Role 1	Role 2	Role 3	Role 4
Role Name	Cost Center 110-120 SVS Role	Cost Center 310 SVS Role	Cost Center 400 SVS Role	Cost Center All SVS Role
Role Code	CC_110_ 120_SVS_ROLE	CC_310_SVS_ROLE	CC_400_SVS_ROLE	CC_ALL_SVS_ROLE
Role Category	Default	Default	Default	Default
Description	Access to cost centers 110 to 120.	Access to cost center 310.	Access to parent cost center 400 and all its children.	Access to all cost centers.
			children.	



The following figure shows the Create Role page for the first role.

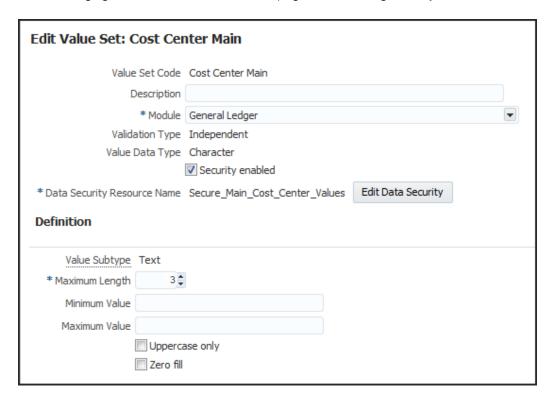


Enabling Segment Value Security for the Value Set

- 1. In the Setup and Maintenance work area, search for and select the Manage Segment Value Security Rules task.
- 2. In the Value Set Code field, enter Cost Center Main and click Search.
- 3. In the Search Results section, click Edit to open the Edit Value Set page.
- 4. Select the **Security enabled** option.
- **5.** In the **Data Security Resource Name** field, enter Secure_Main_Cost_Center_Values.
- 6. Click Save.



The following figure shows the Edit Value Set page after enabling security for the Cost Center Main value set.



Defining the Conditions

Use conditions to specify the segment values that require security.

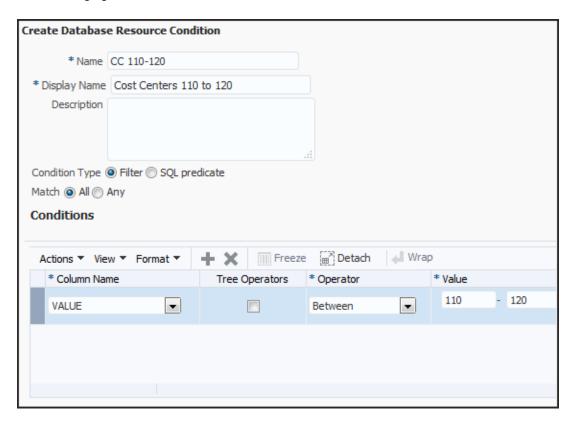
Segment value security rules that provide access to all segment values, and segment value security rules that provide access to single nonparent segment values, don't need a condition. Instead, you can define the policy to cover all values, and you can define a policy to cover a single nonparent segment value provided that you know the internal ID for that segment value. If you don't know the internal ID, you can create a condition for that single segment value.

In this scenario, the internal ID for segment value 310 isn't known, so the following steps create all of the conditions, except for the access to all cost centers, which the policy definition can cover.

- 1. Click Edit Data Security to open the Edit Data Security page.
- 2. On the Condition tab, click **Create** to open the Create Database Resource Condition window.
- 3. Enter CC 110 120 in the Name field.
- 4. Enter Cost Centers 110 to 120 in the **Display Name** field.
- Accept the default value of All for the Match field.
 Matching to all conditions means that all conditions apply simultaneously. Matching to any condition means that any of the conditions would apply.
- 6. Click **Add** in the Conditions section.
- 7. Select VALUE for the Column Name field.
- 8. Select Between for the **Operator** field.
 - Note: You can select one of the following operators: Equal to, Not equal to, Between, Is descendant of, Is last descendant of.
- 9. Enter 110 in the left **Value** field and 120 in the right **Value** field.



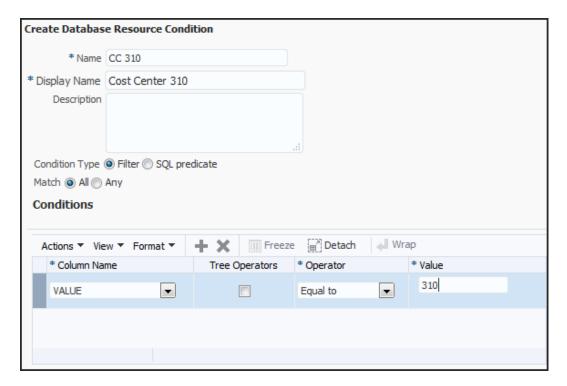
The following figure shows the definition of the first condition.



- 10. Click Save.
- 11. To create the next database resource condition for segment value 310, click **Create** on the Condition tab.
- 12. Enter CC 310 in the Name field.
- **13.** Enter Cost Center 310 in the **Display Name** field.
- **14.** Click **Add** in the Conditions section.
- **15.** Select VALUE for the **Column Name** field.
- **16.** Select Equal to for the **Operator** field.
- 17. In the Value field, enter 310.



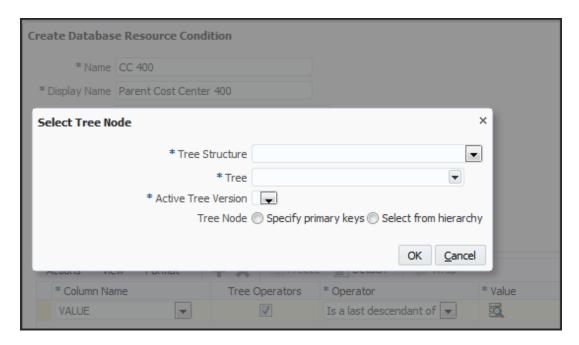
The following figure shows the definition of the second condition.



- 18. Click Save.
- 19. To create the next database resource condition for parent value 400, click Create on the Condition tab.
- 20. Enter CC 400 in the Name field.
- **21.** Enter Parent Cost Center 400 in the **Display Name** field.
- 22. In the Condition section, click Add.
- 23. Select VALUE for the Column Name field.
- **24.** Select the **Tree Operators** option.
- 25. For the **Operator** field, select Is a last descendant of, which restricts access to the parent cost center 400 and all of its children, including intermediary parents.
 - Note: For the Tree Operators field, you can only select Is a last descendant of or Is a descendant of.
- 26. In the Value column, click the Select Tree Node icon to open the Select Tree Node window.



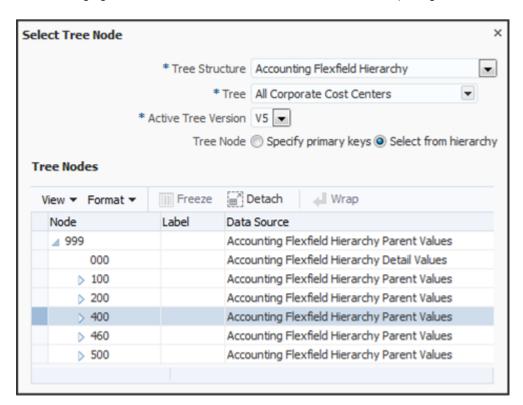
The following figure shows the Select Tree Node window.



- 27. In the **Tree Structure** field, select Accounting Flexfield Hierarchy. This signifies that you are choosing among trees that are used as accounting flexfield, or charts of accounts, hierarchies.
- **28.** In the **Tree** field, select All Corporate Cost Centers.
- **29.** In the **Active Tree Version** field, select V5.
- **30.** In the **Tree Node** field, select the **Select from hierarchy** button. The Tree Node section opens.
- **31.** In the Tree Node section, expand the nodes and select 400.



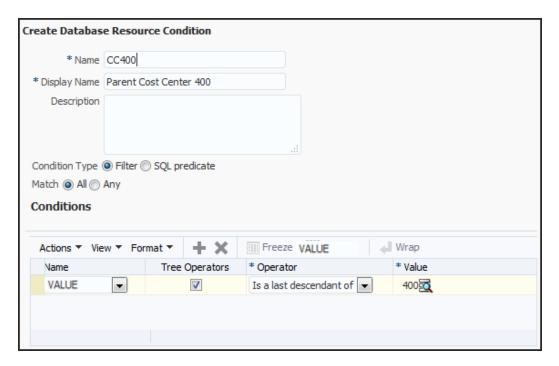
The following figure shows the Select Tree Node window after completing the fields.



32. Click OK.



The following figure shows the definition of the third condition.

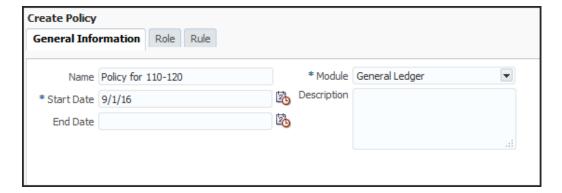


33. Click Save.

Defining the Policies

Create policies to assign conditions to segment value security roles.

- 1. On the Edit Data Security page, click the Policy tab.
- 2. Click **Create** to open the Create Policy window.
- 3. On the General Information tab, enter Policy for 110-120 in the Name field.
- 4. Accept the default value of General Ledger in the **Module** field.
- Enter 9/1/16 in the Start Date field.
 The following figure shows the General Information tab on the Create Policy page for the first policy.



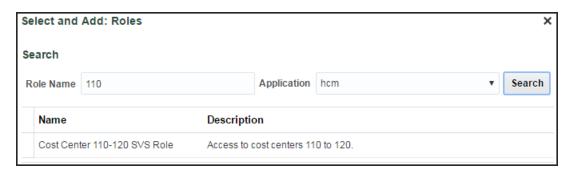
- 6. Select the Role tab and click Add to open the Select and Add window.
- 7. Enter 110 in the Role Name field.
- 8. Select hcm in the **Application** field.



Roles with the Default category are created in the hcm application.

9. Click Search.

The following figure shows the Select and Add window after the search.



Select Cost Center 110-120 SVS Role and click **OK**.
 The following figure shows the Role tab on the Create Policy page for the first condition.



- 11. Select the Rule tab.
- 12. Accept the default setting of Multiple Values in the Row Set field.
 - Note: The Row Set field determines the range of value set values affected by the policy.
 - o If Multiple Values is selected, a condition must be specified.
 - If All Values is selected, then the policy grants access to all values in the value set and no condition is needed.
 - If Single Value is selected, then the internal Value ID for the segment value must be specified and no condition is needed.
- 13. Click Search on the Condition field.
- 14. Select Cost Centers 110 to 120 for the Condition field and click OK.



The following figure shows the Rule tab on the Create Policy page for the first policy.



- 15. Click Save and Close.
- 16. Click **OK** to confirm.
- 17. Repeat steps 2 through 13 to create the rest of the policies, using the values in the following table.

Field	Policy 2	Policy 3	Policy 4
General Information tab, Name	Policy for 310	Policy for 400	Policy for all cost centers
General Information tab, Start Date	9/1/16	9/1/16	9/1/16
Role tab, Role Name	Cost Center 310 SVS Role	Cost Center 400 SVS Role	Cost Center All SVS Role
Rule tab, Row Set	Multiple Values	Multiple Values	All Values
Rule tab, Condition	Cost Center 310	Parent Cost Center 400	

18. Click Done.

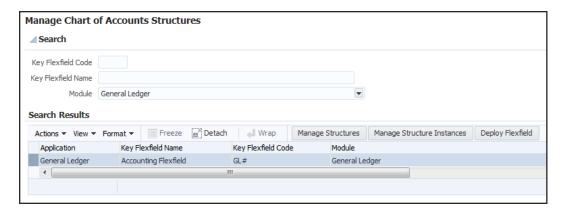
Deploying the Accounting Flexfield

You must deploy the accounting flexfield for the segment value security changes to take effect.

- 1. In the Setup and Maintenance work area, search for and select the Manage Chart of Accounts Structures task.
- 2. In the Module field, select General Ledger and click Search.
- 3. Select the row for the Accounting Flexfield and click **Deploy Flexfield**.



The following figure shows the Manage Chart of Accounts Structure page with the Accounting Flexfield row selected.



4. Click OK.

Publishing the Account Hierarchies

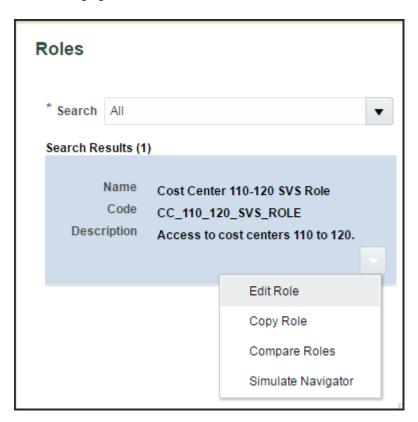
- 1. In the Setup and Maintenance work area, search for and select the Publish Account Hierarchies task.
- 2. In the **Hierarchy** field, select All Corporate Cost Centers.
- 3. In the **Hierarchy Version** field, select V5.
- 4. Click Search.
- 5. In the Search Results section, expand the hierarchy row.
- 6. Select the row for the hierarchy version V5.
- 7. Click Publish.
- 8. Click OK.

Assigning Segment Value Security Roles to Users

- 1. In the Tools work area, open the Security Console.
- 2. Enter Cost Center 110-120 SVS Role in the **Search** field and click **Search**.
- 3. In the Search Results section, select the down arrow icon and select **Edit Role**.



The following figure shows the search results for Cost Center 110-120 SVS Role on the Roles page.



- **4.** Click **Next** four times to navigate to the Edit Role: Users page.
- 5. Click Add User.
- **6.** Enter Casey in the **Search** field and click **Search**.
- 7. Click **Add User to Role** to add Casey Brown to the role.
- 8. Click **OK** to confirm.

The following figure shows the Edit Role page with the user Casey Brown added to the role.



9. Repeat steps 2 through 8 to add the other roles to different users as needed.



Difference in Data Security for GL Features Directly and Indirectly Based on the Balances Cube

When a user is assigned multiple data access sets for the same balances cube with different security specifications for ledger or primary balancing segment value access, a difference is manifested in the data security for those GL features based directly on the cube and those that are not.

General Ledger features based directly on the balances cube are:

- Inquire on Detail Balances
- Account Monitor
- Account Inspector
- Financial Reporting
- Smart View
- Allocations

All other General Ledger features are indirectly based on the balances cube.

- When working on features not directly related to the balances cube, you select a specific data access set and you
 only work with that one data access set at a time. The defined ledger and primary balancing segment value access
 for the selected data access set are enforced
- When working directly with the balances cube, the cumulative effects of your combined data access sets for that cube are enforced. From your combined data access sets of that cube, balances cube security separately constructs the access filter for the ledger dimension and primary balancing segment values dimension independently of the other dimensions. This means the specific combination of ledger and primary balancing segment values access as defined in each distinct data access set are not enforced as such. Instead, you have access simultaneously to all the ledgers and all the primary balancing segment values granted to you through your combined data access sets.
- Note: Balances cube security grants access to all values of the balancing segment value set for a data access set defined as either of the following:
 - Full ledger
 - All Values: Specific Balancing Segment Values Access Type

With segment value security rules assigned to you through your various roles, the security rules are in effect simultaneously whether working directly or indirectly with the balances cube.

Segment value security rules are specified for a particular value set. Therefore, as you are working on anything that references the secured value set, all segment value security rules for that value set that are assigned to you through any of your roles are in effect at the same time, regardless of the specific role the rule was assigned to or the particular role that you are working with at the moment. In other words, segment value security rules are cumulative or the union of all the segment value security rules you have assigned to your through your roles. If you have one role assigned to your user that only grants access to cost center 200, and another role that grants access to cost centers 300 through 500, then you can access to cost centers 200 and 300 through 500.

When working on features not directly based on the balances cube, such as journal entry pages, the primary balancing segment values you can access are based on the intersection of:

Primary balancing segment values granted to you through your current selected data access set.



 All your assigned segment value security rules pertaining to the primary balancing segment value set across all your assigned roles.

So if a balancing segment value is only granted in either of the selected data access set or a segment value security rule, this balancing segment value is not available to you.

In contrast, for features directly based on the balances cube, your access is based on the cumulative union of:

- Primary balancing segment values granted to you through all your assigned data access sets related to the balances cube you are working with.
- Any segment value security rule grants to that primary balancing segment value set across all your role assignments.

Example

In contrast with the preceding discussion about using separate segment value security roles for segment value security rule assignments, the following example shows the data access set and segment value security rules assignments both going to the same role. This setup is used to more easily illustrate the difference in security behavior for features directly and indirectly related to the balances cube.

You are assigned the DAS1 and DAS2 roles below with data access sets that have the following primary balancing segment value specifications.

Role	Data Access Set	Primary Segment Value Assigned
DAS1	Data Access Set 1	01
DAS2	Data Access Set 2	02
DAS3	Data Access Set 3	03

You are also assigned the following primary balancing segment values through a segment value security rule with these same roles.

Role	Primary Segment Value Assigned
DAS1	01
DAS2	03
DAS3	02

Select Data Access Set 1

- 1. For features not directly based on balances cube: You can access Primary Balancing Segment 01 which is the intersection of values from:
 - Data access set for Role DAS1.
 - Security rules grants for Roles DAS1 and DAS2.
- 2. For features directly based on balances cube: You can access Primary Balancing Segments 01, 02, and 03 which are the union of values from data access set and security rules for Roles DAS1 and DAS2



Select Data Access Set 2

- For features not directly based on balances cube: You can't access any Primary Balancing Segment value because there is no intersection of values from:
 - Data access set for Role DAS2.
 - Security rules grants for Roles DAS1 and DAS2.
- 2. For features directly based on balances cube: You can access Primary Balancing Segment 01, 02, and 03 which are the union of values from data access set and security rules for Roles DAS1 and DAS2.

FAQs for Manage General Ledger Security

When does security take effect on chart of accounts value sets for balances cubes?

For new security policies to be effective, the security policies must be defined before the account hierarchies are published to the cube. When you create segment value security rules or change an existing rule that is based on a hierarchical filter, you must republish the tree version. Use the Publish Account Hierarchies page to republish the tree version and for the security to become effective.

Note: Changes to an account hierarchy previously published to the balances cube require that the hierarchy be republished to the cube to reflect the updated hierarchy.

What happens when changes are made to an account hierarchy that is referenced in segment value security rules?

The tree is set from an active to a draft state. The rules referencing the account hierarchy become ineffective.

After making changes to your hierarchy, you can submit the Process Account Hierarchies process to automatically run the required steps for processing account hierarchies updates in one submission, including:

- Tree audit
- Tree activation
- Row flattening
- Column flattening
- Maintain value set
- Maintain account hierarchy
- Publish hierarchy

With a successful audit process, the hierarchy is set back to an active status. The rules referencing the account hierarchy go back to being effective using the updated hierarchy.



Run the row and column flattening processes for the updated hierarchy as the flexfield component in the application as well as other hierarchy processes rely on the flattened hierarchy data to come up with the list of values available to the user to properly secure the correct account values.

Run the Maintain Value Sets and Maintain Chart of Account Hierarchies processes, particularly for hierarchy changes to the primary balancing segment value set if such values are referenced in your primary balancing segment value based data access sets. These processes update the data that is required to regulate ledger and data access security by storing:

- Primary balancing segment values assigned to a ledger.
- Specific child balancing segment values assigned to a data access set through parent value assignments.

How can I secure the data in GL balances cubes?

Use data access set and segment value security to secure dimension values such as ledger and chart of account values. For chart of accounts dimension values, security restricts the display of data associated with the secured values, but not the selection of the values themselves. For example, when submitting a report, you can select company value 100 in your report definition when selecting the Point of View, even if you weren't granted access to that company value. However, you can't see the data associated with company 100 in your report.

Manage Reporting Currencies

Reporting Currency Balances: How They Are Calculated

Reporting currency balances, set at the journal or subledger level, are updated when General Ledger journals are posted and converted to your reporting currencies. This process includes:

- General Ledger manual journals, periodic journals, and allocations.
- At the subledger level, journals from Oracle Fusion Subledger Accounting.
- Other journals imported from sources other than your Oracle Fusion subledgers.

When you post a journal in a ledger that has one or more reporting currencies defined, the posting process:

- Creates journals converted to each of your reporting currencies.
- Includes them in the same batch as the original journal with a status of Posted.

Settings That Affect

Reporting currencies share a majority of the ledger options with their source ledger. For example, the reporting currency uses the same suspense account and retained earnings accounts as its source ledger. However, there are certain options that must be set specifically for reporting currencies, such as the currency conversion level. The currency conversion levels are Balance, Journal, and Subledger.

Note: Secondary Ledgers cannot use subledger level reporting currencies.



Multiple dependencies exist between a reporting currency and its source ledger. Therefore, complete your period opening tasks, daily journal or subledger level reporting currencies accounting tasks, and period closing tasks in the correct order. Some guidelines are presented in the table below.

Туре	Task		
Period Opening Tasks	Open the accounting period in both your ledger and reporting currencies before you create or import journals for the period. Converted journals are only generated in your reporting currency if the period is open or future enterable.		
Daily Tasks	Enter the daily conversion rates to convert your journals to each of your reporting currencies.		
Period Closing Tasks	 Finish entering all regular and adjusting journals for the period in your ledger. Post all unposted journals in your ledger if not already done in the previous step. Post all unposted journals in your reporting currencies if not already done in the previous step. Run Revaluation in both your ledger and reporting currencies. Post the resulting revaluation batches in each ledger. As needed, translate balances in your ledger. Generate needed reports from both your ledger and reporting currencies. Close your accounting period in both your ledger and reporting currencies. 		

How Reporting Currencies Are Calculated

If you use reporting currencies at the journal or subledger level, journals are posted in your reporting currency when you:

- Create accounting.
- Post journal entries.
- Translate balances.

General Ledger and Subledger Accounting automatically generate journals in your reporting currencies where the entered currency amounts are converted to the reporting currency amounts. Other factors used in the calculation of reporting currency balances are listed:

- Manual Journals: Enter a manual journal batch in your reporting currency at the journal or subledger level by using
 the Create Journals page. Select the journal or subledger level reporting currency from the ledger's list of values.
 Continue in the same manner as entering any other manual journal.
- Conversion Rounding: Use the reporting currency functionality to round converted and accounted amounts using the same rounding rules used throughout your Oracle Fusion Applications. The reporting currency functionality considers several factors that are a part of the currencies predefined in your applications, including:
 - Currency Precision: Number of digits to the right of the decimal point used in currency transactions.
 - Minimum Accountable Unit: Smallest denomination used in the currency. This might not correspond to the precision.
- Converted Journals: Generate and post automatically journals in your reporting currencies when you post the original
 journals in the source ledger for the following types of journals:
 - Manual journals
 - Periodic and allocation journals
 - Unposted journals from non-Oracle subledger applications



- Unposted journals from any Oracle Fusion subledger that does not support reporting currency transfer and import
- Optionally, revaluation journals
- Unconverted Journals: Rely on the subledger accounting functionality to converted and transfer Oracle Fusion subledger journals for both the original journal and the reporting currency journal to the General Ledger for import and posting. The reporting currency conversion for these journals is not performed by the General Ledger.
- Approving Journals: Use the journal approval feature to process reporting currency journals through your
 organization's approval hierarchy. You can enable journal approval functionality separately in your source ledger and
 reporting currencies.
- Document Numbers: Accept the default document numbers assigned by the General Ledger application to your
 journal when you enter a journal in your ledger. The converted journal in the reporting currency is assigned the same
 document number. However, if you enter a journal in the reporting currency, the document number assigned to the
 journal is determined by the reporting currency.
- Sequential Numbering: Enable sequential numbering to maintain the same numbering in your reporting currency and source ledger for journals, other than those journals for Oracle Fusion subledgers. Do not create separate sequences for your reporting currencies. If you do, the sequence defined for the reporting currencies is used. The sequences can cause the document numbers not to be in sync between the ledger and reporting currencies.
 - Note: General Ledger enters a document number automatically when you save your journal if:
 - o The Sequential Numbering profile option is set to Always Used or Partially Used.
 - Automatic document numbering sequence is defined.

If you use manual numbering, you can enter a unique document number.

- Revaluation: Run periodically revaluation in your ledger and reporting currencies as necessary to satisfy the accounting regulations of the country in which your organization operates.
- Account Inquiries: Perform inquires in the reporting currency. You can:
 - o Drill down to the journal detail that comprises the reporting currency balance.
 - Drill down to see the source ledger currency journal amounts from any automatically converted journal that was created when the original journal posted.
- Note: Be careful when changing amounts in a reporting currency, since the changes are not reflected in your source ledger. Making journal entry changes to a reporting currency makes it more difficult to reconcile your reporting currency to your source ledger. In general, enter or change your journals in your source ledger, and then allow posting to update the reporting currency.
- Note: If you use journal or subledger level reporting currencies, statistical journals are generated for your reporting currencies, but the journals are not converted.

Manage Ledger Sets



Ledger Sets: Examples

When you use multiple ledgers, you can group all types of ledgers in a ledger set. For example, group primary ledger, secondary ledgers, and reporting currencies (journal and subledger levels) as long as they share the same chart of accounts, calendar and period type.

Ledger sets are used to manage ledgers, including:

- Opening and closing of periods.
- Running reports and processes for multiple ledgers simultaneously.
- Supporting adjustments and allocations.

Regulation Requirements Example

You have 26 registered companies in Vision Corporation. The regulations require that each company is to maintain a separate ledger. All companies share the same chart of accounts and calendar.

You set up a ledger for each company and group them into a ledger set. You can then use the ledger set to perform all accounting activities, while the data remains partition for reporting purpose for each company.

Multiple Countries Example

You have a parent company in one country. You operations in five other countries. Each country has its own legal entity and must report to both the parent company and the local government. All legal entities use the same chart of accounts and calendar. Currency is different and Reporting Currency ledgers have been created. You can group all of the primary and reporting currencies ledgers into one Ledger Set. You can then perform all accounting activities as though you had one worldwide ledger.

Manage Accounting and Reporting Sequences

Accounting and Reporting Sequences: Overview

When you record your journals, you can assign unique sequence numbers. Sequence numbering the journals simplifies the task of tracing the journal entries and is a legal requirement in some countries, especially those in Europe, Asia, and Latin America. Sequencing enables the fiscal authorities to easily verify the completeness of a company's accounting record.

Sequential numbering of journals enables you to:

- Maintain gapless sequence numbers.
- Sequentially number subledger journal entries. Generate a sequence number for journal entries created by subledger accounting before transferring them to general ledger.
- Sequentially number general ledger journal entries.
- Assign sequences within a ledger or legal entity.
- Generate reports based on the sequence numbers.



The two different types of journal sequences used with Subledger Accounting and General Ledger journals are:

- Accounting Sequences: Gapless sequence numbering for journals which is assigned when the journals are posted
 or subledger accounting runs.
- Reporting Sequences: Gapless and chronological sequence numbering which is assigned to journals when the General Ledger period is closed.
- Note: You can sequence journals for ledgers (primary and secondary ledgers) and journal level and subledger level reporting currencies as well as legal entities. If you use journal or subledger level reporting currencies, first define sequences for the source ledger. The sequence number generated for the source ledger is not automatically assigned to the reporting currency journal. You must then set up accounting and reporting sequences separately for the reporting currencies.

Related Topics

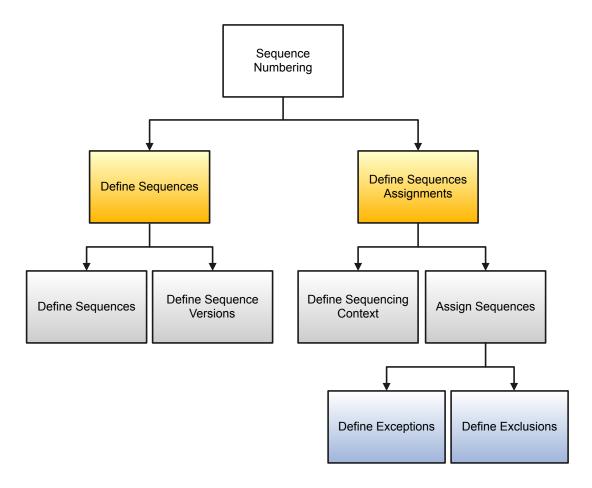
Create Chart of Accounts, Ledger, Legal Entities, and Business Units in Spreadsheets: Explained

Defining Journal Sequences Manually: Explained

Define sequences manually to generate numbers for journal entries. Use the Manage Accounting and Reporting Sequences task to search, create, and update sequences, sequence versions, and sequencing assignments. You can also create journal sequences using the Rapid Implementation process. The process creates and assigns the sequences.



The following figure explains the process involved in sequence setups, which includes defining and assigning sequences.



Sequences

Create, update, and search for sequences and sequence versions in the Sequences tab on the Manage Accounting and Reporting page.

- Define Sequences: To define a sequence, enter a unique sequence name and sequence description.
- **Define Sequence Versions**: Create at least one sequence version to complete the sequence. A sequence version indicates the series of sequence numbers to be used and the effective date range of the series.

Field Name	Description
Status	Displays the status of the sequence. The status could be New, Used, and Disabled.
Effective Start Date	The effective start date uses the current system date, but can be set to any date in the past or future.
Effective End Date	The date when the sequence should be disabled.
Initial Number	An integer that is greater than zero. The number is incremented by one for each accounting entry.



Field Name	Description
Last Number Used	Last generated sequence number for the version.

Sequencing Assignments

After creating your sequences, assign them with various criteria to specify when the sequences are generated. Use the Sequencing Assignments tab on the Manage Accounting and Reporting page to search, update, or create sequencing assignments. The top region of the tab contains the fields that make up the Sequencing Context definition.

The following table explains the fields used.

Field Name	Description
Ledger Name	The ledger for which the journal entries must be sequentially numbered or that contains the legal entity.
Legal Entity	The legal entity for which the journal entries must be sequentially numbered. The Legal Entity field appears when you set your Sequence By ledger option to Legal Entity.
Journal Entry Type	Select whether general ledger or subledger entries are being sequentially numbered.
Sequencing Event	The event which triggers the sequence numbering,
Sequence Validation Date	The date when generating a sequence number. This date is used to determine if a sequence is available and the sequence assignment is active.
Require Assignment for All Journals	If you select this option, you must explicitly define valid Sequence Assignments, Exceptions, or Exclusions for all journal entries.
Status	Indicates if the assignment is enabled or disabled.

The following table outlines some of the different combinations of Sequencing Contexts.

Journal Entry Type	Sequencing Event	Sequence Validation Date	Journal Sorting Date	Remark
General Ledger	Posting	Accounting Date	Accounting Date	Accounting Sequence
Subledger Accounting	Subledger Accounting	Accounting Date	Completion or Posting Date or Accounting Date	Accounting Sequence
General Ledger	Period Close	Accounting Date	Accounting Date	Reporting Sequence
General Ledger	Period Close	Reference Date	Reference Date	Reporting Sequence
Subledger Accounting	Period Close	Accounting Date of Reference Date	Reference Date	Reporting Sequence



	Journal Entry Type	Sequencing Event	Sequence Validation Date	Journal Sorting Date	Remark
--	--------------------	------------------	--------------------------	----------------------	--------

Tip: For reporting sequences, you can specify Fiscal Balancing Segment Values to specify what balancing segment values you can use with reporting sequences for period close. You can deselect the option to exclude balancing segments with management type journals or nonfinancial journals such as statistical journals.

Fiscal Balancing Segment Values		
Segment Value	Description	Fiscal
30	Italy	✓
31	Italy Management	V
32	Italy Fiscal	V

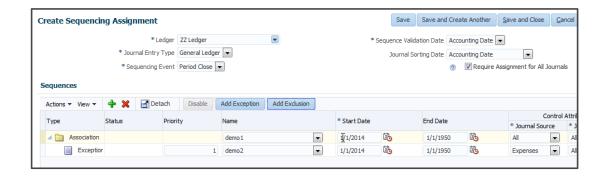
Sequencing Assignment Details

The following table explains the fields in the Sequences region on the Create Sequencing Assignment page.

Field Name	Description	
Status	Status of the assignment such as New or Used.	
Priority	Priority of the exception. This option is inactive unless you define an exception.	
Sequence Name	Name of the sequence to be assigned. A blank indicates the entries must not be sequenced.	
Start Date	Define the effective start date for the sequence assignment.	
End Date	Define the effective end date for the sequence assignment.	
Control Attributes		
Journal Source	Select the journal sources from the list to be sequentially numbered for a sequence event.	
Journal Category	Select the journal categories from the list to be sequentially numbered for a sequence event.	
With Subledger		
Accounting Event Type	Select the accounting event types to be sequentially numbered (only for subledger entries).	
Accounting Entry Type	Select the accounting entry types to be sequentially numbered (only for subledger entries).	
Document Category	Select to group transactions assigned to the sequence.	



Use the following buttons to add a row:



- Add Exception: When you create an exception, the priority value is always one more than the highest number in the
 Priority field. Use exceptions to assign different sequences to certain journal sources, journal categories, accounting
 event types, or accounting entity types. For example, you want to include all journal sources, but assign a different
 sequence for your Expense journals.
- Add Exclusion: Create exclusions to exclude specific journal sources, journal categories, accounting event types, or accounting entity types. For example, you want to exclude journal sources and categories containing management or statistical accounts that do not impact financial results.

FAQs for Manage Accounting and Reporting Sequences

What happens if I enable sequencing in the rapid implementation spreadsheet?

Two sets of defaults are generated: general and country specific. The process applies the general defaults first and the country specific defaults override general defaults. For example, using the predefined defaults, sequencing is enabled at the ledger level. For some countries like Spain and Italy, the general defaults are overridden by country defaults and sequences are set at the legal entity level.

Accounting and reporting sequences are generated based on the predefined country defaults. For example, for a France ledger, accounting sequences are generated for each legal entity of the ledger. For an Italian ledger, both accounting and reporting sequences are generated for each legal entity of the ledger.

Manage Journal Approval Rules

Creating Journal Approval Rules

Watch: This video tutorial shows you how to create journal approval rules that either automatically approve a journal batch or that route the batch for supervisory approvals based on the ledger and journal amounts.



Approving Journals: Points to Consider

Oracle Fusion Applications journal approval uses Oracle Fusion Approvals Management (AMX) to merge the functionality of Oracle Approvals Management (AME) and Oracle PeopleSoft Approvals (AWE). In addition, Oracle Business Process Execution Language (BPEL) has replaced Oracle Workflow.

Rule Definition Consideration

One predefined approval rule exists for journal approval. If you enable the ledger and the source for approval, then the journal entry is sent for one level of approval. Configure the approval rules in the AMX Rules Setup user interface. For a simple approval scenario, start by defining one or all of the following journal approval rules based on:

- The highest journal line amount per ledger per batch.
- The highest journal amount per ledger per batch.
- Where you are in the period close process. For example, are you in the beginning, middle, or end of the month, or in preclose, close, post close, or quarter close process?

Select the **Skip creator for Approval List** option on the journal approval rules configuration page to prevent the submitter from approving a journal batch. If the journal batch submitter is found to be one of the approvers then you can select to skip the journal submitter as an approver. Then assign the approval task to other approvers or automatically route the approval task to the submitter's manager.

Create the following rules to apply when your maximum journal line amount is:

Rule Condition	Approval Action
Less than 50,000 USD	No approval required
Between 50,000 to 100,000 USD	Requires one level of approval
Greater than 100,000 USD	Requires two levels of approval

Build your rules for combinations of ledger, entered amount, approval level, or other scenarios. In addition, define your rules based on attributes from the different parts of your journal, including the ledger, batch, header, or line level. For example, use category, source, account, or descriptive flexfield information as selection criteria for journal approval.

AMX List Builder Considerations

Use the following AMX List Builders to build your approval list.

List Builder	Functionality	Additional Information
Human Resources (HR) Supervisory	The HR Supervisory hierarchy levels are selected. The number of levels available for approval is specified.	The most effective when the General Accountant enters the journals. For example, if an accountant enters a journal, he needs approval from his manager. If his manager enters a journal, he needs approval from his manager, and so on up the hierarchy for the



List Builder	Functionality	Additional Information specified number of levels. Self-approval can be set at any level in the hierarchy.
Job Level	A relative dollar amount is attached to a job. The approval list moves up the HR Supervisory hierarchy to the point where approval finds a job with the necessary approval amount.	Enable self-approval to allow approval of journals created within your authority limit.
Position	A relative dollar amount is attached to a position.	Use this hierarchy if you need: A hierarchy different than the HR Supervisory hierarchy Multiple hierarchies that must be selected based on different attributes.
Approval Group	Approver groups represent functional or subject matter experts outside the transaction's managerial chain of authority, such as Legal or Human Resource personnel.	
Dual Chain	Dual chains can be processed at the same time.	

Note: Best practices: Select the Job Level, HR Supervisory, or Position list builder for your journal approval rules.

Other Considerations

Other functionality to consider before defining approval rules.

- Approval is for the entire journal batch regardless of the attributes used in the approval rules.
- For the job and position level approvals, the approval list continues up the hierarchy until it finds the approver with the correct approval authority.
- If the journal requires approval, submitting a journal for posting automatically routes the journal for approval before posting.
- A journal can be escalated to an approver by the administrator.
- The **Withdraw Approval** button on the Journals page is used anytime in the approval process to withdraw journals from the process. Clicking this button enables editing of the journal. After your changes are made, submit the entry for approval again. When a journal is withdrawn, the completion status is set to Incomplete.
- Approval notifications display a table of key journal attributes for each journal and a list of past, current, and future approvers.
- The Journals region of the dashboard displays journals requiring your approval and journals pending approval from others.
- If you are the current approver, the Journals page shows the journals to be approved or rejected.
- Allocation journals are not routed through the approval process.



- You can review the details of the journals and journal lines included in a journal batch on the online and e-mail journal batch approval notifications.
- Note: Approval is enabled at the ledger and source level. Both the ledger and journal source must be enabled for the approval process.

Related Topics

Approval Management: Highlights

Manage AutoPost Criteria Sets

Creating an AutoPost Criteria Set: Worked Example

This example shows how to create an AutoPost Criteria Set to post your general ledger journal entries that were created by the journal import process for your subledger transactions. Your enterprise, InFusion Corporation:

- Implemented Oracle Fusion General Ledger and the Oracle Fusion subledgers: Payables and Receivables.
- Uses a non-Oracle subledger called Fast Assets for fixed asset tracking and depreciation.
- Plans to automate posting of your general ledger journal batches created by the journal import process to protect
 the subledger sourced journal entries from edits or deletion that might cause an out-of-balance situation between
 your subledgers and general ledger.

Consider the following points while creating your criteria set:

- Use the **All** option for category and accounting period to reduce maintenance and ensure that all journal imports are included in the posting process.
- Create a criteria set that includes all your subledger sources. Create multiple criteria sets by source only if you need to schedule different posting times to balance close activities or reduce processing time.

Creating an AutoPost Criteria Set

Create your AutoPost Criteria Set to automatically post journal entries from both Oracle and non-Oracle subledgers.

- 1. On the Manage AutoPost Criteria Sets page, click the **Create** icon to open the Create AutoPost Criteria Set page.
- 2. Enter the set name: All Journal Imported Entries
- 3. Select the **Enable** check box.
- 4. Enter the description: Posting journals imported from the subledgers.
- 5. Click the **Add Row** icon to add each new line.
- **6.** Complete the fields, as shown in the table below:

Priority	Ledger or Ledger Set	Source	Category	Accounting Period
1	InFusion Corporation Ledger	Payables	All	All
2	InFusion Corporation Ledger	Receivables	All	All



Priority	Ledger or Ledger Set	Source	Category	Accounting Period
3	InFusion Corporation Ledger	Fast Assets	All	All

7. For all three sources, select **Yes** for the **process all criteria** option and enter 30 as the number of days before and after submission date.

Setting the before and after days with a wide range of days enables the process to run less often.

- 8. Click the Save and Close button.
- **9.** Schedule the process to run daily at 3:00 a.m.

Schedule the process immediately after the journal imports to prevent changes to the journals. Run the process during nonpeak times to save resources.

Manually Generating the AutoPost Process: Examples

Create an AutoPost criteria set and schedule the AutoPost process to run on a regular basis following your scheduled journal imports from your subledgers. When errors occur that prevent posting of the journal imports, you must correct the errors and manually run the AutoPost process. The following scenarios illustrate the kinds of errors that could occur and how you can resolve these errors.

Scenario

The following errors occurred and prevented the journal batches from posting when the scheduled AutoPost process ran.

Error	Cause	Solution
Error - Unopened accounting period	The journal import was imported into a future period. An error arises when the AutoPost process runs on a schedule because journals cannot be posted in a future period.	Open the period.
Error - Invalid or no journals	Journal import fails to import transactions from the general ledger interface table. The AutoPost process runs on schedule but finds no batches to post. The Posting process does not run and the AutoPost Execution report shows that no batches matched the criteria.	Correct the error that caused the journal import to fail.
Error - Invalid or no journals	No journals were selected based on the posting criteria. Journal batches are available for posting. The Posting process does not run and the AutoPost Execution report shows that no batches matched the criteria.	Revise the criteria set.

After you correct the errors:

 Manually run the AutoPost process by selecting the Launch AutoPost option from the Tasks panel on the journal pages.



• By clicking the **Generate** button on the AutoPost criteria set pages. Verify that the process ran successfully by reviewing the AutoPost Execution report.

FAQs for Manage AutoPost Criteria Sets

How can I run the AutoPost process?

After you define an automatic posting criteria set, run the AutoPost process by clicking the **Generate** button on the **Manage AutoPost Criteria Sets** page or the **Launch AutoPost** link from the **Journals** task pane. The AutoPost process posts the journal batches that meet the criteria defined. Optionally, schedule the AutoPost process for specific automatic posting criteria sets through the Schedule tab in the Schedule Process: Advanced region to run at specific times and submission intervals.

How can I identify errors that occurred during my AutoPost process?

Review the AutoPost process results on the AutoPost Execution report. This report is automatically created when the process completes successfully. The report contains the batch name, accounting period, and balance type for each posted journal batch, and lists error statuses for batches that fail to post. The unposted journals with their error status are also displayed on the Requiring Attention tab of the Journals work area and the General Accounting Dashboard.

Why didn't the AutoPost process post journal batches as expected?

Verify that the posting criteria set specifies the precise criteria required to post the wanted journals. If the criteria is correct, then verify the following:

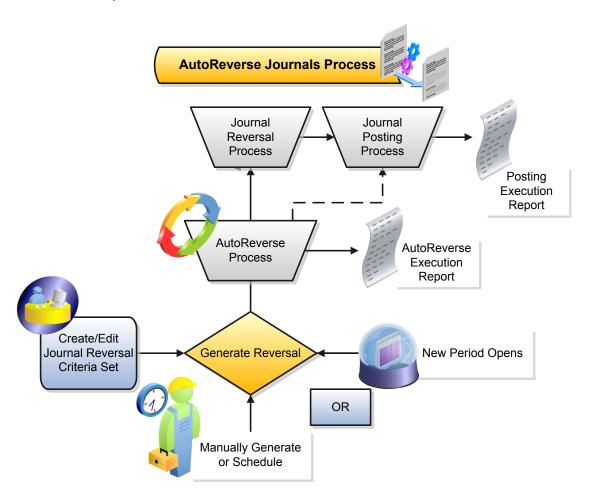
- Journal imports completed successfully.
- Journal batches are error free and ready to post.
- Specified accounting period is open.

Manage Journal Reversal Criteria Sets



Automatic Journal Reversals: How They Are Processed

The ability to submit journal reversals automatically enables you to automate and streamline your journal reversal process. If you generate a large number of journal reversals for month end closing, use the automatic reversal functionality to save time and reduces entry errors.



Settings That Affect Journal Reversals

The journal must meet the following criteria to be automatically reversed:

- Balance type is Actual.
- · Category is enabled to be automatically reversed.
- Reversal period is open or future enterable.
- Posted but not yet reversed.
- Not a reversal journal. Reversal journals cannot be reversed in Oracle Fusion General Ledger.
- Not a posted journal for a reporting currency that was replicated from its source journal. Reporting currency journals
 that were replicated from a source journal are reversed when the source journal is reversed.
- Not a posted journal that originated from Oracle Fusion Subledger Accounting with a frozen source.



A new ledger option called **Launch AutoReverse After Open Period** enables generation of automatic journal reversals when an accounting period is first opened. This ledger option replaces the former profile option called GL: Launch AutoReverse After Open Period. If you reverse your journals on the last day of every month, disable the ledger option to automatically run reversals when the period is opened. Then schedule the AutoReverse process to run on the last day of every month. The accounting period automatically increments for each subsequent run.

How Automatic Journal Reversals Are Processed

Define Journal Reversal Criteria Sets to automatically reverse and optionally post journals using the following criteria:

Criteria	Functionality	Options
Category	Required. The journal category you set as the reversal option. Journals entered with this category are selected for reversal and optionally,	All journal categories are listed.
Reversal period	posting. Required.	No default Same period
	The accounting period of the reversal journal. The Next day option is only applicable to average daily balance ledgers. Nonaverage daily balance ledgers and consolidation average daily balance ledgers treat the Next day option in the same manner as the No default option.	Next periodNext nonadjustingNext day
Reversal day	Required for average daily balance ledgers only. The day of the period on which to reverse the journal.	First dayLast dayNext day
Reversal method	Required. The method for changing the amounts in the reversal entry.	Change signSwitch debit or credit
Automatic reversal option	Required. The option to reverse and post journals automatically. Journals are posted after they are reversed.	NoneReverse automaticallyReverse and post automatically

After creating your journal reversal criteria sets, assign them to ledgers. Journal reversal criteria set can be shared and assigned to multiple ledgers. Also secure journal reversal criteria set definitions using definition access set security to prevent unauthorized users from using, viewing, or modifying the journal reversal criteria.

If the automatic reversal option is set to reverse and post automatically, the AutoPost process posts the journals generated by the AutoReverse process. The process does not post other journals. You manually post reversal journals that were generated outside of the AutoReverse process.



Note: Journals posted by the AutoReverse process always bypass approval.

General Ledger automatically creates the AutoReverse Execution report when the AutoReverse process completes successfully. The report prints the journal name and reversal period for each journal that is successfully reversed and whether the reversal journal is submitted for posting. The AutoPost Execution report is created automatically when the AutoPost process finishes. These reports help you diagnose any problems and verify that all journals were processed properly.

Note: The AutoReverse process does not check that the reversal date is a valid business day for an average balance ledger. The journal validation in the journal pages or import process does the check and if necessary, rolls the date to the next business day.



5 Define Period Close Components

Manage Period Close

Opening First Period: Overview

For all ledgers, primary, secondary, and journal and subledger level reporting currencies, open the first period of the ledger when you are ready to transact in that period.

- 1. To open the first period of your ledgers, navigate to the **Open First Period** task in the primary ledger task list.
- 2. Click the Go to Task icon.
- 3. On the submission page, select the ledger and the period to open.
- 4. Click the **Submit** button to submit the open period process.

You can use other ways to open the first period or subsequent periods without going into the **Setup and Maintenance** work area. You can maintain the ledgers' period statuses from the:

- Close Status region in the **General Accounting Dashboard**. The **Close Status** region provides real-time visibility into the period close process from your subledgers to your General Ledger across the entire enterprise.
- Manage Accounting Periods task in the Period Close work area.
- Process Monitoring work area, which provides a framework for submitting, monitoring and maintaining processes across Oracle Fusion Financials.

Close Monitor: Overview

The Close Monitor:

- Provides information on the period close status for a given accounting period across multiple products for related ledgers in a hierarchical ledger set based display.
- Uses the hierarchical ledger set to mirror the consolidation relationships and roll ups of entities across the enterprise.
- Summarizes period close status information for each ledger across multiple products and for each consolidation node across multiple ledgers.
- Provides the contact information of the manager for a given node on the ledger set hierarchy.
- Summarizes high level income statement results for each entity and aggregates this financial information at each consolidation node.
- Displays each of these elements of information, period status, manager information, and financial data, in separate tags that are navigated to for each node of the interactive hierarchical display.
- Provides views for a given ledger set, for a particular accounting period, and currency.

The period status information that is displayed is broken down by application module including General Ledger, Payables, Receivables, Asset, Projects, and Costing. Some modules track their entity at a more granular level, such as:

- Business units for Payables, Receivables, and Projects
- Asset Books for Assets



Cost Organization Books for Costing

The Close Monitor indicates the number of the subunits by module for the ledgers. It also displays the fractional indicator, where applicable, of how many of the subunits are at the closed status.

Secondary ledgers, journal level, or transaction level reporting currencies cannot be associated with subledger business units for Payables, Receivables, and Projects. As such, if the ledger set displayed in the hierarchy includes members that are secondary ledgers, journal, or subledger level reporting currencies, the period status indicated in the Close Monitor for such subledger modules is based on its related primary ledger. Asset books and cost organization books can be associated with all types of ledgers. Therefore in the case of the Assets and Costing modules, their period status for secondary ledger or reporting currencies is shown accordingly for the books directly associated with them. Otherwise, their period statuses are derived from the books associated with their primary ledgers.

Setting Up the Close Monitor

The Close Monitor setup is comprised of a ledger set hierarchy definition whereby a predefined ledger set is addressed, with each ledger and ledger set assigned a manager who is responsible for its financial close, and a logo to represent the entity in the display.

Note: The list of managers available for assignment contains the persons defined in the Human Capital Management (HCM) module of Oracle Fusion Applications. The attributes defined in HCM, such as the picture of the person and contact details, are shown in the Close Monitor.

The ledger set serves as the foundation of this setup.

- The members of the Close Monitor hierarchy must share a common chart of accounts and calendar.
- The financial data displayed in the Close Monitor is derived from the account group assigned to the ledger set, therefore, an assignment is required. The account group:
 - Must include two line items whose account designations respectively query the total revenues and total expenses of the organization.
 - o Reflects a summarized income statement in the financial data tab of the Close Monitor.
- All ledgers in the ledger set share a common chart of accounts and the selection of accounts are equally applicable throughout the nodes in the ledger set hierarchy.
- When working with ledger sets that include members that are also ledger sets, you can choose any of the ledger sets in the selector to indicate the top starting ledger set to display in the Close Monitor.
- If different account groups are assigned to each ledger set member in such a ledger set, the account group used to display the financial data is the one assigned to the ledger set specified in the selector in the Close Monitor.
- To have meaningful comparison and summation along the ledger set hierarchy:
 - Assign ledgers to the ledger set that have a relevant currency representation that matches the intended group currency that the Close Monitor displays the financial data in.
 - Select the appropriate primary, secondary, or reporting currency ledger for assignment to the ledger set.
 - Alternately, use translated balances (balance level reporting currency) in the ledger set selection to satisfy the common group currency requirement if needed.

Viewing the Close Monitor

You choose a ledger set, an accounting period, and currency as the view criteria for the Close Monitor display. You can alter this selection at any time.



For example, change the currency displayed by:

- Working with a global ledger set.
- Shifting the focus to a lower level ledger set that is aggregating at the continental level, such as North America, that uses a different group currency.
- Including the ledger with the relevant currency representation that matches the selected group currency that the Close Monitor financial data is displayed in.
- Note: If matching financial data for a ledger in the selected currency is not available, a message is displayed stating that the requested financial data is not available.

The Close Monitor supports different zoom levels to enable you to:

- Accommodate viewing a larger ledger set hierarchy in its entirety, given the limited display area of the user interface.
- Show detail information for each node which can vary, decreasing and simplifying in content as you zoom out further to be able to accommodate showing more nodes in a single view.
- Hover over the more summarized node and view a punch out of that particular node that shows the complete set of information available at the 100% zoom level.
- Leave the zoom level at 100% and move around the display to other ledger sets or ledgers currently not in view.
- Note: A view control panel that can be exposed on demand allows you to adjust the zoom level, pan across the hierarchy, flip the display tabs, and switch the hierarchy display format.

Configuring Social Objects in Oracle Social Network: Explained

Before you can start using the social object, for example, accounting period, journal, or intercompany transaction in Oracle Social Network, configure the social object using the Manage Oracle Social Network Objects task on the Setup and Maintenance task list page.

The configuration consists of enabling the social object and its attributes for use on the Oracle Social Network. For example, for the accounting period social object, enable the following attributes: Ledger, Period Name, Period Start Date, and Period End Date. You also configure the enablement method of the social object. The methods are: No, Manual, and Automatic.

The configuration applies to all instances of that social object in the application and to all ledgers. You can automatically create an conversation by setting the option in **Managing Oracle Social Network Objects** user interface.

Note: Oracle Social Network is currently only available in Oracle Cloud implementations.

Related Topics

Managing Oracle Social Network Objects: Explained

Period Close Components: Explained

While implementing your accounting configuration, optionally define and maintain the period close components to customize your accounting configurations setup.

Period close components include allocations, period entries, revaluation, and historical rates.



If you use allocations, revaluation, or translation, configure the following tasks under the **Define Period Close Components** parent task in your implementation project:

- Manage Allocations and Period Entries
- Manage Revaluations
- Manage Historical Rates

Manage Allocations and Period Entries

Manage Allocations and Period Entries is a manual task in the implementation project. Use the **Calculation Manager** to create allocations and other formula journal templates for generating periodic journal entries automatically. Base formulas on multiple criteria.

You must perform an external procedure outside the Setup and Maintenance work area to complete this task. To set up your allocations rules, navigate to the Journals work area and click the **Create Allocations Rules** task from the **Tasks** pane. This task navigates you to **Calculation Manager**, a framework that enables you define your allocation rules and formulas using a graphical interface and intuitive step-by-step wizards.

Manage Revaluations

Defines currency revaluation options, such as the range of accounts to revalue and the gain or loss accounts. Revaluation is done to adjust foreign entered amounts due to currency fluctuations. Navigate to the Manage Revaluations page, and define and generate your revaluation definitions.

Manage Historical Rates

Historical rates are the weighted average rate for transactions that occur at different points in time. Used by the application to calculate the conversion rate on equity account balances during foreign currency translation of the balance sheet.

Navigate to the **Currency Rates Manager** page to define and maintain your historical rates that are used in the translation process. In Oracle Fusion General Ledger, you can currently define historical rates using an ADF Desktop Integrator spreadsheet.

To create historical rates, specify the required Ledger and the other optional fields, as needed. Click the **Create in Spreadsheet** button to open the spreadsheet for uploading.

To update the existing historical rates for your ledgers, click the Edit in Spreadsheet button, the spreadsheet is prepopulated with the existing historical rates.

Note: Before using the historical rates spreadsheet, install the ADF Desktop Integrator client as an add-on to Microsoft Excel.

FAQs for Manage Period Close

How can I use social networking to effectively close the period?

Use the Social link on the **Period Close** work area to collaborate with members of your team or others within your company to effectively close the period.



For example, as a controller, you keep **Oracle Social Network** open from the **Period Close Overview** page during the period close so you can be aware of any transactions that must be posted for the period.

On the All tab:

- You see a conversation that needs your attention.
- Your boss, the chief financial officer, started a private conversation with you to announce the close of a deal worth 15,000,000 USD and wants it booked for this period.
- You download and listen to a voice message file that the chief financial officer posted sharing details about the delivery of the goods to help you confirm that the revenue can be posted to this period.
- You create a new conversation and invite your accounting manager to join, marking it so she knows to reply quickly.
- Your accounting manager added you to a conversation for the revenue adjustment journal.
- Your accounting manager adds a post to the conversation confirming that the revenue is posted.

You navigate to the Close Monitor page to view the latest financial balances and confirm that the revenue is posted.

Depending on your job role and permissions, you can access social networking features for the following Oracle Fusion General Ledger business activities:

- Period status
- Journal

Related Topics

What does social networking have to do with my job?

Manage Allocations and Periodic Entries

Creating an Allocation Rule and Generating Allocations

Watch: This video tutorial shows you how to create an allocation rule and generate allocation journals.

Allocation and Periodic Entries: Overview

In Oracle Fusion General Ledger, use the Calculation Manager to create allocations and other formula journal templates for generating periodic journal entries automatically. Allocations are defined and generated from preaggregated balances in the GL Balances cubes, which provide the following benefits:

- Immediate real-time access to financial balances for allocations and periodic entries.
- Accelerated performance for complex allocations.

You can base formulas on multiple criteria. For example:

- Use account balances or statistical amounts to allocate shared revenue or costs across multiple organizational units and ledgers.
- Define complex computations based on variables from different charts of accounts.



Group journal formulas together and execute sequentially to update account balances in a step-by-step process.

The Calculation Manager provides flexibility, automation, intelligence, and control in distributing costs and revenues across the enterprise. In addition, the Calculation Manager:

- Includes run time variables, rules, formulas, and rule sets stored in Oracle Essbase.
- Distributes revenues or costs with recursive allocation rules.
- Creates complex formula rules using formula components.
- Contains an Allocation Wizard to define allocation and formula rules.
- Uses real-time checking of rule definitions to validate correctness of rules.
- Minimizes setup and maintenance time with reusable components.
- Simplifies allocation generation mechanism by integrating with enterprise schedule.
- Groups rules together in rule sets and cascading allocations for processing efficiencies.
- Creates primary, statistical, or foreign currency allocation and formula rules.

Access the Calculation Manager from the **Tasks** pane of the **General Accounting** dashboard or **Journals** work area by clicking the:

- Define Allocation Rules link to define or modify allocation definitions
- Generate Allocations link to run the allocation process
- Note: Adobe Flash Player 10 or above is a required component for the Calculation Manager. Upgrade your Adobe Flash Player if the Calculation Manager's performance slows down after upgrading your browser. For more information, see:
 - Designer's Guide for Oracle Hyperion Calculation Manager

Calculation Manager: Overview

The Calculation Manager creates, validates, deploys, and administers sophisticated allocation rules. In the Calculation Manager:

- Base formulas on multiple criteria, such as account balances or statistical amounts, to allocate shared revenue or costs across multiple organizational units.
- Use complex computations based on different variables to automatically calculate allocated amounts.
- Group journal formulas together and executed sequentially to update account balances step-by-step.

There are three types of objects that can be created in Calculation Manager:

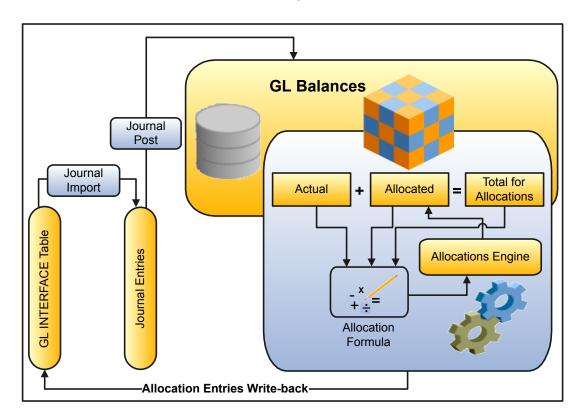
- Components: Contain formulas, points of view, or allocation objects.
- Rules: Contain components such as points of view, formulas, and templates, which are used to create allocation calculations.
- Rule Sets: Contain sets of rules that can be calculated sequentially



- Note: The following are limitation in Oracle Fusion General Ledger.
 - Allocation rules cannot be shared across rule sets in Calculation Manager.
 - Within a rule or rule set, the same target or offset cannot be written to by multiple rule components.
 - When generating allocation rules with run time prompts other than the User Point of View in an allocation rule component, an error occurs.

Oracle Essbase Balances Cubes: Overview

Oracle Essbase is embedded within Oracle Fusion General Ledger and provides multidimensional balances cubes. Every time a transaction or journal is posted in General Ledger, the balances cubes are updated at the same time.



The flowing table lists the Essbase Dimensions and examples of dimension members.

Dimension	Description	Example
Accounting Period	Based upon the calendar of the ledger or ledger set. Report on years, quarters, or periods.	2012QTR-1JAN-12
Ledger or Ledger Set	Used to select a ledger for the reporting. Multiple ledgers may be in the same cube if they share a common chart of accounts.	InFusion North America Ledger SetInFusion US Primary Ledger



Dimension	Description	Example
Chart of Accounts Segments	Uses a separate dimension for each of the segments from the charts of accounts. Organized by hierarchy. A default hierarchy is provided that includes all detail segment values. Hierarchies published in the Publish Account Hierarchies user interface are included.	 Company: InFusion America: 101 Cost Center: Sales: 400 Account: Cash: 1110
Scenario	Indicates if the balances represented are actual or budget amounts. Allocation-related dimensions are predefined members and required for allocation solutions. Allocation dimensions are not used directly by end users. Budget scenario dimension members are user-defined in the Accounting Scenario value set and appear in the cube after running Create Scenario Dimension Members process.	Budget 2012ActualsForecast 2013
Balance Amount	Indicates if the value is the beginning balance, period activity, or ending balance. Debit, Credit, and Net amounts are available for reporting.	 Beginning Balance (DR, CR, or Net) Period Activity (DR, CR, or Net) Ending Balance (DR, CR, or Net)
Amount Type	Indicates whether the amounts represent Base, Period to Date, Quarter to Date, or Year to Date.	BasePTD: Period to DateQTD: Quarter to DateYTD: Year to Date
Currency	Used to select the wanted currency for the balances.	All ISO CurrenciesUSD: US DollarJPY: Japanese Yen
Currency Type	Used to select the currency type of the balances.	TotalEnteredConverted From (for each ISO currency)

Allocation Security: Explained

To access the Calculation Manager, you must be assigned one or more of the four privileges and permissions.

The following privileges and permissions are associated with the Calculation Manager:

- Generate General Ledger Allocation Formula Generate Allocation and Periodic Entries: Permits generation of allocation and periodic entries.
- Define General Ledger Allocation Formula Manage Allocation Rules or Rulesets through Calculation Manager:
 Grants the ability to update allocation rules or rule sets owned by the user with view access to all allocation rules or rule sets regardless of their ownership.



- Define Self Managed General Ledger Allocation Formula Manage Allocation Rules or RuleSets through Calculation Manager: Grants the ability to update allocation rules or rule sets, but limited to the ones owned by the user.
- Administer General Ledger Allocation Formula Administer Allocation Rules or RuleSets through Calculation Manager: Grants the ability to update all aspect of allocation rules or rule sets including the ownership attribute, regardless of the original definition's ownership.

Migrating Allocation Rules: Instructions

Migrating Allocation Rules consists of two tasks:

- 1. Exporting from the Source Instance
- 2. Importing to the Target Instance

Exporting from the Source Instance

To export from the source instance:

- 1. Sign in to the Oracle Fusion Home page of the source environment with Application Implementation Consultant.
- 2. Navigator > Setup and Maintenance.
- 3. Click Manage Implementation Projects in the Task pane.
- 4. Click the **Create** icon to add an implementation project.
- 5. Enter the basic information and click **Next**.
- **6.** On the **Create Implementation Project: Select Offerings to Implement** page, verify that the **Include** check box is deselected for all the projects.
- 7. Click Save and Open Project.
- 8. Click the **Select and Add** icon to add a task.
- 9. Change Search to Tasks and search for Manage Allocations and Periodic Entries.
- 10. Select Manage Allocations and Periodic Entries and click Apply. The task is added to the Implementation Project page in the background.
- 11. Click Done.
- **12.** Click **Done** for the implementation project.
- 13. In the Tasks pane, click Manage Configuration Packages.
- **14.** Click the **Create** icon to create a configuration project.
- 15. Search for your implementation project in the **Name** field and enter or modify the basic information.
- **16.** Select the option **Setup task list and setup data**.
- 17. Click **Next** and then **Submit**.
- **18.** Answer Yes to the message.
- 19. Click **Refresh** until the process finishes.
- 20. Click the **Download** icon, select the download Configuration Package, and save it to a local disk.

Importing to the Target Instance

This process imports the allocations and periodic entries.

To import to the target instance:

- 1. Sign in to the Oracle Fusion Home page of the target environment with Application_Implementation_Consultant.
- 2. Navigator > Setup and Maintenance.
- 3. Click Manage Configuration Packages.
- 4. Click **Upload** to upload the configuration package that was downloaded in the export process.
- 5. Click Browse to find the file, then Get Details, and then Submit.
- 6. In the Export and Import Processes table at the bottom of the page, click Import Setup Data to import the data.
- 7. On the **Import Setup Data** page, accept or change defaults as needed.



- 8. Click Next to navigate through the pages and then Submit.
- 9. Click **Refresh** until the submitted process finishes.

Manage Recurring Entries

Recurring Journals: Overview

Define recurring journal formulas for transactions that you repeat every accounting period, such as accruals, depreciation charges, and allocations. Your formulas can be simple or complex. Each formula can use:

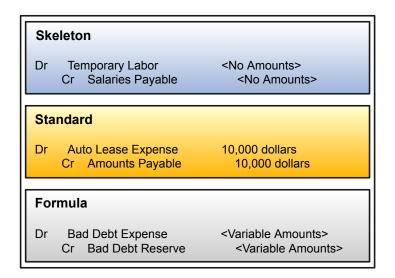
- Fixed amounts and account balances, including standard, actual amounts, statistics, and period-to-date or year-to-date balances.
- Amounts from the current period, prior period, or same period last year.
- Amounts in your formulas, including total balances, entered currency balances, or statistical balances.

You can quickly create recurring formulas by copying and modifying existing formulas. You can:

- Define single ledger or multiple ledger recurring journal formula batches.
- Create a recurring journal formula batch that contains recurring journal entries for different ledgers.
- Define recurring journal formulas for your ledger currencies, entered currencies, and statistical currency.

Recurring Journal Types: Explained

You normally use three types of recurring journal entries to reduce data entry time and increase accuracy for journal entries that repeat each period.



Skeleton Journal Entries: Contain the same accounts each period, but have different amounts. After you generate
skeleton journal entries, edit the unposted journal batch by entering the journal line amounts on the Edit Journals
page.



Use skeleton journal entries to record statistical journals, such as headcount, units sold, barrels of oil, or other statistical factors. For example, to enter headcount for your cost centers each period:

- Define a skeleton entry with your headcount accounts.
- Generate the skeleton entries.
- Enter the actual headcount amounts before posting the batch.
 - Note: Set the journal entry to reverse automatically at the beginning of the next period if you enter the total headcount each period. Otherwise, if you only enter the change in the headcount each period, a reversing journal is not required.

Best practices recommend that you create skeleton recurring journal entries in spreadsheets or copy existing journals.

To create journals in spreadsheets:

- Navigator > Journals.
- Select the Create Journal in Spreadsheet link to download the workbook template once.
- Create and save the skeleton journal entry.
- Each period open the template and enter the amount for the journal lines already in the template.
- Upload the batch.
- Open the journal in the Edit Journal page and add the amounts.

Once the updates are made, save, complete, and post the journal batch.

To copy journals:

- Navigator > Journals > Manage Journals.
- Search for the journal you want to copy.
- Open the journal.
- Click on the Batch Actions Menu > Copy.
- Make wanted changes to the new journal.
- Save, complete, and post the journal batch.
- 2. Standard Recurring Journal Entries: Contain the same accounts and amounts each period. Just as with skeleton recurring journal entries, best practices recommend that you create standard recurring journals in spreadsheets.
 - Navigator > Journals.
 - Select the Create Journal in Spreadsheet link to download the workbook template once.
 - Create and save the standard journal entry.
 - Each period, upload, and submit the batch with posting selected.

The recurring journal batch is created and posted.

3. Recurring Journal Formula Entries: Contain formulas created using the formula component and allocation wizard in the Calculation Manager. These formulas calculate journal amounts that vary from period to period and are based on existing account balances or other criteria.



Use recurring journal entries to perform simple or complex allocations or eliminations. For example, you can allocate:

- A portion of your rent expense to another division.
- A pool of marketing costs to several departments based on the ratio of department revenues to total revenues.

Creating Recurring Journals: Example

This example shows how to define and generate formula recurring journals that are automatically generated every period.

You must have:

- A role that can access the Journals work area in Oracle Fusion General Ledger.
- A duty that can access the Create Allocation Rules task.

Assumptions

- The chart of accounts includes segments for company, department, account, subaccount, and product.
- Account 1210 is the trade receivables account.
- The PTD period activity in account 1210 is 100,000 USD.

Goals

- The goal is to create journal that populates a monthly allowance for bad debt based on 5% PTD activity in the trade receivables account.
- Account 7730 is the bad debt expense account and account 1260 is the allowance for bad debt account.
- A formula rule must be defined to generate the following journal entry for the period Apr-11 and thereafter.
 - DR 01.000.7730.0000.000 5,000 USD
 - o CR 01.000.1260.0000.000 5,000 USD

Definitions

- Configuration: Create a formula rule to achieve the above goal.
- Create the Run-Time Prompt Variable: Create an RTP variable as an optional component of a rule. When you generate an allocation based on a rule with a defined RTP variable, you are prompted to specify a dimension member for the RTP. The variable is use in the allocation calculation.
 - For example, use an RTP variable of Accounting Period, which prompts you to specify the period to use in the allocation calculation. An RTP variable can be created once and used in multiple rules.
- Create the Rule Set: Create a rule set. Rule sets are created by combining two or more related rules together to enable sequential allocating of balances.
- **Generate Allocation Journals**: Start the allocation process to create the journal entries that populate the account balances.

Configuration

- 1. Navigate to the Journals work area.
- 2. Click the Create Allocation Rules link on the Tasks panel.



- 3. Navigate to the Administer menu option and then Calculation Manager. Calculation Manager opens in a new browser window and a cube is highlighted based on the data access set selected in the Journals work area.
- 4. Expand Essbase.
- 5. Expand VF_USA_Accounting_Flexfield (your cube).
- 6. Expand db.
- 7. Highlight the Rules row, right click, and select New from the menu.
- 8. Enter the Rule Name: Special Bad Debt Allocation, accept the other defaults, and click OK button.
- 9. The Rule Designer opens in a new tab. Under New Objects, click, hold, and drag the Point of View object. Place it between the Begin and End nodes in the Rule Designer.
- 10. Enter a Caption: Point of View.
- 11. Perform the following steps to enter a Variable Dimension Value:
 - a. Click the Value field for Accounting Period.
 - b. Click the Actions icon and select Variable from the drop-down list. A new window opens.
 - c. Under Category, select Database from the drop-down list.
 - d. Click Accounting_Period.
 - e. Click OK button.
- **12.** Perform the following steps to enter Other Member Dimension Values:
 - a. Click the Value field for another dimension.
 - **b.** Click the Value field for another dimension.
 - c. Click the Actions icon and select Member from the drop-down list.
 - **d.** Select a member and click on the blue select arrow pointing right.
 - e. Click the OK button. Repeat for all dimensions to include in the Point of View.

In this scenario, the following are fixed dimension values:

- Ledger: Vision Operations (USA)
- Company: 01Department: 000Subaccount: 0000
- Product: 000Currency: USD
- Currency Type: Total
- **f.** Under New Objects, click, hold, and drag the Formula component. Place it between the Point of View nodes in the Rule Designer.
- g. Enter a Caption: Bad Debts Calculation.
- h. Enter the Offset member.
- i. Click Next button.

In this scenario, the offset is defined as account 1260, the allowance for bad debt. The offset is child combination 01.000.1260.0000.000 when combined with the fixed member dimension values in the Point of View.

13. Perform the following steps to enter the Formula Member Dimension Value:

In this scenario, the formula member dimension value is defined as account 7730. The bad debt expense is charged to child combination 01.000.7730.0000.000 and combined with the fixed member dimension values in the Point of View.

- a. Click the icon for the formula field and select Member from the drop-down list.
- b. Select the Account dimension value, highlight the row, and click the blue select value pointing right.



In this scenario, the goal is to calculate an allowance for bad debt based on the PTD period activity in trade receivables account 1210. Trade receivable is child combination 01.000.1210.0000.000 when combined with the fixed member dimension values in the Point of View.

c. Repeat for the other formula member values and click the OK button when all formula members are selected.

In this scenario, the following dimension values are selected. Selection of members for the dimensions below is required for the source in a formula component.

Scenario: Actual

Balance Amount: Period Activity

Amount Type: PTD

- **d.** Multiply the formula expression by .05.
- e. Click the Save icon.
- f. Click the Validate and Deploy icon.

Create the Run-Time Prompt Variable

- 1. Navigate to the Journals work area.
- 2. Click the Create Allocation Rules under Tasks.
- 3. Once the Calculation Manager opens in a new browser window, a cube is highlighted based on the data access set selected in Journals work area. To define the run time prompt, select Variables under the Tools menu.
- 4. Expand to the db under the cube, highlight the row, right-click on the row, and select New from the menu.
- 5. The Variable Designer opens in a new tab. Enter the variable header and value information.

A default value must be entered and the variable name cannot contain any spaces.

Variable	Header Information
Name	Accounting_ Period
Туре	Member

Variable	Value Information
Dimension	AccountingPeriod
Default Value	Apr-11
RTP	<checked></checked>
RTP Text	Enter Accounting Period

6. Click the Save icon. The RTP variable is ready for use.

Create the Rule Set

- 1. Navigate to the Journals work area.
- 2. Click Create Allocation Rules under the Tasks pane.



- 3. Once the Calculation Manager opens in a new browser window, expand to Rule Sets under the highlighted cube, highlight the row, right-click on the row, and select New from the menu.
- 4. Enter the rule set name and click the OK button.
- 5. The Ruleset Designer opens in a new tab. Expand to the db under the cube for which the rule set is created, expand the rules, and drag the rules under the rule set.
- 6. Click on the row for the rule set, click the Variables tab, and check Merge Variables.

Merge Variables means that common variables among all of the rules in the rule set are merged. You only have to select the run-time prompt value once when submitting the Generate Allocations process.

- 7. Click the Save icon.
- 8. Click the Validate and Deploy icon.

Generate Allocation Journal

- 1. Navigate to the Journal work area.
- 2. Click Generate Allocations under Tasks.
- 3. Select a rule or rule set and enter any run-time prompt values.
- 4. Uncheck the Post Allocations check box if automatically posting the generated allocations is not wanted.
- 5. Click the Submit button.
- 6. Generate Allocations submits four processes consecutively (three if the Post Allocations check box is not selected). Those processes calculate the allocation, write back the results to the GL_INTERFACE table, import the journal batches, and posts the journal batches to the Oracle Fusion General Ledger.

Scheduling Recurring Journals: Examples

You can create processing schedules for recurring journal entries that have been defined in the **Calculation Manager**. Scheduling automates the monthly generation of the entries and speeds up the close process.

You can define multiple schedules for each calendar in General Ledger. These schedules can be increment by accounting period based on any calendar defined. Schedules are shared across ledgers.

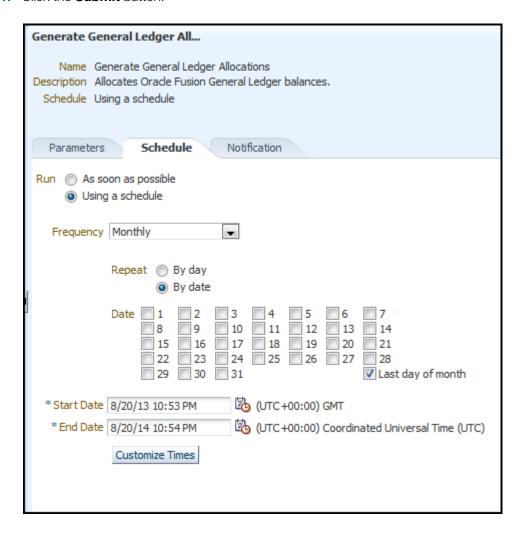
Scenario

In this example, you have created a reserve for bad debt recurring journal entry in the **Calculation manager**. Now, add a recurring schedule to the entry to generate the entry once a month on the last day.

- 1. Navigator. The Generate Allocations > Journals > Generate General Ledger Allocations page opens.
- 2. Select the Rule or Rule Set: Reserve for Bad Debt.
- 3. Specify Accounting Period: Blank
 - Note: The Accounting Period field appears if you use the Run-Time Prompt in your rule and select Accounting Period as the run-time variable.
- 4. Check Post Allocations.
- 5. Select the **Advanced** button.
- 6. Select the **Schedule** tab.
- 7. Click Using a schedule.
- 8. Select **Frequency**: Monthly.
- Select Repeat: By Date.
- 10. Enter start and end dates.



11. Click the Submit button.



12. The generation process waits in the **Schedule Processes** page until the schedule time, which in this example is the last day of the current month.

Manage Allocation Entries

Calculation Manager Toolbar: Explained

In addition to the Oracle Hyperion Enterprise Performance Management Workspace buttons, the Calculation Manager toolbar displays buttons that are specific to the Calculation Manager. Not all buttons display in all the views and designers within the Calculation Manager.

The Calculation Manager toolbar consists of the following buttons:

- · Home: Displays the default startup option for the content area.
- System View: Displays the main view within the Calculation Manager. (This is the default view).



- List View: Displays a list of objects that you can filter by application type, application, object or database type, deployment status, and validation status.
- Custom View: Displays a view you can customize with folders you create and objects you drag and drop into them.
- Filter Options: Opens the Filter dialog that you can use to filter objects in the List View.
- Refresh: Refreshes the view with your latest changes.

The Calculation Manager toolbar adds the following buttons when you open a rule:

- Save: Saves the object with which you are working.
- Validate: Validates the object with which you are working.
- Validate and Deploy: Validates and deploys the object with which you are working.

Calculation Manager Menus: Explained

Calculation Manager menus and menu options display in addition to Oracle Hyperion Enterprise Performance Management Workspace menus and menu options. The menus and options vary depending on the view you are using and the object with which you are working. The default view of the Calculation Manager displays the following menus when you launch Calculation Manager, System View.

Note: This topic describes the Calculation Manager menu options only.

File Menu

Enables you to create new objects, open and close objects, import and export objects, print rules, and log off.

- Note: Not all of these file menu options are available for the products that use Calculation Manager.
- New, Rule: Creates a new rule
- New, RuleSet: Creates a new rule set

Edit Menu

Enables you to edit objects you select. It is available from most of the views and from within the Rule and Component definition pages.

- Edit, Delete: Deletes an object selected in the System, List, or Custom View
- Edit, Copy: Copies selected text
- Edit, Paste: Pastes text copied to the clipboard to the right of the cursor
- Edit, Copy Group: Copies a component group
- Note: The Edit menu is not available within the Deployment View.

View Menu

Enables you to open different views.

• View, View Pane: Displays or hides a list of existing and/or new objects that you can add to rules, rule sets, components, and templates by dragging and dropping them.



- Note: This is the only View menu option available from within the Rule Designer and Ruleset Designer.
- View, List View: Displays a list of the objects you select on the Filter dialog. The filter dialog enables you to create a filtered list, by application type of applications, databases, and objects.
- View, System View: Displays a list of the Essbase applications, databases, and objects to which you have access.
 This is the default view.
- View, Custom View: Displays a view that you can customize with folders you create and drag and drop objects into them. This view enables you to organize objects in a way that is meaningful to you.
- View, Deployment View: Displays a list, by application type and application, of the rules and rule sets that are deployed and not deployed with their deployment and validation status.

Tools Menu

Enable you to install other products, search for objects, create a filtered list of objects for the List View, edit the caption of an object, and access the Variable Navigator and Variable Designer.

- Tools, Filter: Opens the Filter dialog from which you can filter by application type, application, object type (rule, rule set, formula or script component, or template), calculation type, plan type, database, deployment status, and validation status. You can also select All to display all application types, applications, objects, and databases, regardless of their deployment and validation status.
- Tools, Variables: Opens the Variable Navigator in which you can navigate to a location for which you want to create, edit, copy, or delete a variable. From the location you select in the Variable Navigator, you can display the Variable Designer in which you can create, edit, copy, and delete variables for components.
 - Note: The two menu options listed above are not available within the Deployment View.

Actions Menu

Enables you to validate and deploy objects you select in the views and from within the Rule and Ruleset Designers. Not all of the Actions menu options are available from within the views and designers.

- Actions, Validate: Validates the rule, rule set, and formula component you selected
 - Note: This is the only Actions menu option available from within the Deployment View.
- Actions, Deploy: Deploys the rules or rule sets you selected.
- Actions, Quick Deploy: Deploys the rule in fewer steps than regular deployment by using a shortcut to one or more applications.
 - Note: This feature is available only from within the Rule Designer for Essbase business rules.



Using Flow Charts: Explained

View rules and templates, and the components that comprise them, in a flow chart within the Rule Designer. When you open a rule, move among the components that comprise it, for example, formulas, ranges, and loops, by selecting them in the flow chart. Increase or decrease the size of the flow chart to view or hide details of the components.

When you select a component in the flow chart, its properties, usages, and other information are displayed in tabs below the flow chart. As you move among the components, the tabs below the flow chart change. For example, if you open an allocation rule and select the formula component in the flow chart the following properties are displayed:

- Properties of the formula, such as name, description, application, and application type to which the formula belongs
- Usages of the formula, such as which rules and templates the formula is used in.
- Text of the formula, such as the variables, members, and functions, that are displayed in the tabs below the flow chart.

Views: Explained

Views enable you to see Calculation Manager objects in different contexts. For example, the Deployment View displays objects according to whether they are deployed or not deployed. The Custom View displays objects according to filters and criteria that you select.

The Calculation Manager contains the following views:

- List View
- System View
- Custom View
- Deployment View
- View Pane

List View

The List View contains a filtered list of Essbase applications or databases, and objects, rule sets, rules, or formula components, filter according to your criteria.

System View

The System View is the default view that is displayed when you start the Calculation Manager. It contains a list of all of the applications and objects to which you have access. Your privileges are determined by the role you are assigned in Shared Services. For each object, the owner, the user who made the last change, and the date the changes were last made are listed.

Custom View

The Custom View enables you to create folders and drag and drop objects into them to create a view that contains only your objects. This view enables you to organize objects in a way that is meaningful to you.



Deployment View

The Deployment View contains a list of the rules and rule sets that are deployable with their deployment and validation status. From this view, select rules and rules sets in an application to make them deployable. Deploy one or more rules or rule sets (partial deployment), or deploy all rules and rule sets in an application (full deployment).

View Pane

The View Pane enables you to create or open an object. Display the View Pane in the left frame of the window. Depending on whether you are working in a rule or a rule set, the Rule or Ruleset Palette, is displayed in the View Pane. In the Palette, drag new and existing objects and drop them into the rule, rule set, or flow chart.

When working with views display or hide the View Pane using the View menu. In the Custom View, drag and drop new and existing objects from the View Pane into the custom folders you create. In the System and List views, the View Pane is hidden by default. In the Deployment View, the View Pane is not available.

Note: The View Pane content varies depending on which view you are in and whether you are working with a rule set, rule, template, or component.

The following table lists the tasks that can be performed from the various views in the Calculation Manager.

Tasks	List View	System View	Custom View	Deployment
Create, open, rename, delete, refresh, and close objects	Yes	Yes	Yes	Yes
Set preferences	Yes	Yes	Yes	Yes
Import and export objects	Yes	Yes	Yes	Yes
Show the usages of objects	Yes	Yes	Yes	Yes
Create a copy of objects	Yes	Yes	Yes	Yes
Print a business rule	Yes	Yes	Yes	Yes
Select views	Yes	Yes	Yes	Yes
Sign out Workspace	Yes	Yes	Yes	Yes
Work with favorites	Yes	Yes	Yes	Yes
Perform an advanced search	Yes	Yes	Yes	Yes
Start help	Yes	Yes	Yes	Yes



Tasks	List View	System View	Custom View	Deployment
Filter objects in the view according to criteria you specify	Yes			
Work with variables	Yes	Yes		Yes
Validate objects	Yes			Yes
Create a shortcut to a business rule		Yes		
Import and export business rules and other objects		Yes	Yes	
Validate and migrate objects		Yes	Yes	
Change the owner of an object			Yes	Yes
Deploy objects				Yes

Filtering Objects in the List View: Examples

You can use filters in the List View to filter objects according to:

- Application Type that is populated by default with the application type, such as Essbase, your application or database, in which you are creating the new rule.
- Object type, such as allocation rules, allocation rule sets, and formula components
- Deployment or validation status

Scenario

To create a filtered list of objects in the List View:

- 1. From the System View, select View, List View.

 The Filter dialog is displayed the first time you open the List View. If you select filtering options, then close the List View to work in the System or Custom View. When you reopen the List View, the filter dialog is not displayed. To change the filtering options when you reopen the List View, select Tools, Filter to open the Filter dialog.
- 2. In the Filter dialog, on Filter Options, under Application Type, select Essbase.
- 3. Do one of these tasks:
 - For Essbase: In the Application and Object Type fields, select the applications and object types you want to display in the List View. The default is All.
 - Under Deployed Status and Validated Status, clear any check boxes of selections you do not want to display.
 All check boxes are selected by default.



- Tip: Click Reset to reset the dialog with default values.
- On Advanced Options, for Object Label, select one of these options to display only objects whose names match the criteria:
 - Starts With, to display only objects whose names start with characters you specify.
 - Ends With, to display only objects whose names end with characters you specify.
 - Contains, to display only objects whose names contain characters you specify.
 - Matches, to display only objects whose names match characters you specify.
- o Enter the characters that are common to the names of the objects you want to display.
- Select Ignore case to display objects whose names contain characters in either upper or lower case, even if the case does not match the case of the text you entered in step 6.
- In Created By, enter the name of the user who created objects you want to display.
- o In Modified By, enter the name of the user who modified objects you want to display.
- For Created Date, select After, Before, or Between to display only objects that were created after, before, or between dates you specify. Between is the default. Click the list of values to display calendars from which you can select dates.
- For Modified Date, select After, Before, or Between to display only objects that were modified after, before, or between dates you specify. (Between is the default.) Click the list of values to display calendars from which you can select dates.
- For Any Text, select an option to display only objects containing text that starts with, ends with, contains, or matches text that you enter. To display objects that include this text regardless of its case, select Ignore case
- 4. Click OK.

FAQs for Manage Allocations and Periodic Entries

How can I access the Calculation Manager?

Login into the Oracle Fusion General Ledger application and navigate to the **Journals** work area. From the **Journals** work area select the **Create Allocation Rules** link and automatically log into the Calculation Manager in Workspace to create new allocation rules or rule sets.

Note: The application or the balances cube that is currently selected in the General Ledger Data Access Set is automatically selected in the Calculation Manager.

How can I create a folder in the Custom View?

In the Custom View, create folders that contain only the allocation rules, allocation rule sets, and formulas you want to view and work with. To add objects to your folders, drag them from the Existing Objects pane and drop them into the folders.

To create a folder in the **Custom View**:

1. In the System View, select View, Custom View.



- 2. In the Custom View, right-click the Essbase application type, and select New Folder.
- 3. In **New Folder**, enter a name for the folder.
- 4. Click OK.
- Tip: You can create nested folders by right-clicking the folder you want to create a folder in and selecting New Folder.

How can I rename a folder in the Custom View?

Rename the folders you create in the Custom View.

To rename a folder in the Custom View:

- 1. In the System View, select View, Custom View.
- 2. In the **Custom View**, expand the **Essbase** application type.
- 3. Right-click the folder you want to rename, and select **Rename**.
- 4. In the **Rename Folder**, enter a new name for the folder.
- 5. Click OK.

Define Allocation Rules

Allocation Rules: Explained

The Calculation Manager enables you to create, validate, deploy, and administer sophisticated multidimensional allocation rules. An allocation rule is logical expressions or formulas that are created within an application to produce a set of resulting values. Create an allocation rule set of two or more related rules to launch the rules sequentially.

Before you create a rule or rule set, you must understand the database outline and the application with which you are working. This information helps you create your allocation rules more efficiently. Learn the following about your data:

- How the data is stored and aggregated.
- What level the data gets loaded into the database.
- What order the calculations are performed.
- What key assumptions drive the calculations.

Create allocation rules using components like formulas, member ranges, and variables, including run time prompt variables.

Creating an Allocation Rule: Example

You can create one or more allocation rules to use to allocate balances, as needed for financial reporting from the **System View**. You can also create an allocation rule from the **List, Custom,** and **Deployment Views**.

Scenario

To create a new rule:

Navigator > General Accounting: Journals > Create Allocation Rules link.



- 2. Navigate menu > Administration > Calculation Manager.
- 3. File menu > New > Rule.
- 4. In **New Rule**, enter the rule's name.
- **5.** Enter the **Application Type**: Essbase.
 - Note: The application type is populated by default with the application type in which you are creating the new rule.
- 6. Select an **Application Name**. The application name must be a valid Essbase application such as your chart of accounts name.
- 7. Select the **Database**.
 - Note: If you expand the following options in the System View: Essbase > your Application > Database name, then right click Rules and select New to create a new rule, the New Rule dialog is populated with the Application Type, the Application, and the Database you are working in within the System View.
- 8. Click **OK**. The new rule is displayed in the **Rule Designer**.

Starting Calculation Manager: Worked Example

This example demonstrates how to begin to create allocations and other formulaic journal templates to generate periodic journal entries automatically.

Starting Calculation Manager

- 1. Navigate to the General Accounting, Journals work area.
- 2. In the General Accounting, Journals work area, click the Create Allocation Rules link.
- 3. In the Enterprise Performance Management System Workspace, select Administer, then Calculation Manager from the Navigate menu.
- 4. In Allocation Manager, select Variables from the Tools menu.
- 5. Expand the Essbase tree.
- 6. Expand the database to which you want to add a rule set. For this example, expand VF_USA_Accounting_Flexfield.
- 7. Right-click the **db** icon and select New from the list.

Note: If the database already has a rule set, you can add a rule by right-clicking the rule set and selecting New.

8. On the Properties tab, complete the following fields.

Field	Description	
Name	Balancing_ Segment_Value	
Туре	Member	
Dimension	Company	
Default Value	3888	
RTP Text	Enter Company Text	



9. Click Save.

Creating an Allocation Rule with an Allocation Component: Worked Example

This example demonstrates how to create an allocation rule with allocation components.

You are the General Accountant for InFusion America Inc. You are creating an allocation rule with allocation components.

A rule consists of calculations that are grouped into components.

When you design allocation rules, you add variables to components. Variables assume the values that you define for them. Replacement variables provide a substitution value when you design or generate an allocation rule.

Creating an Allocation Rule with an Allocation Component

- 1. Click the Navigator.
- 2. Click the **Journals** link under General Accounting to access the Journals work area.
- 3. Click the Create Allocation Rules link.
- In the Enterprise Performance Management System Workspace, click the Navigate menu.
- 5. Select Administer and then Calculation Manager.
- 6. On the System View tab of Calculation Manager, expand the Essbase tree.
- 7. Expand the VF USA Accounting Flexfield tree node.
- 8. Expand the db tree item.
- 9. Right-click the Rules tree item.
- 10. Select New.
- 11. In the New Rule dialog box, enter EMEA Overhead Distribution in the Name field.
- **12.** Click **OK**.

You design a rule by dragging components from the Rule Palette to the Rule Designer flow chart. Each component performs a separate task. You can add Point of View, Allocation, or Formula components to your rule. You enter additional rule details on the Properties tab.

- 13. In the Rule Palette area, drag the Point of View object to the Rule Designer area.
 - The begin and end components of the rule appear in the Rule Designer work area.
- 14. On the Point of View tab in the lower panel of the workspace, click in the **Ledger** field of the Dimension value table.
- **15.** Click the **Actions** button to the left of the **Ledger** field.
- **16.** Select Member.
- 17. In the Member Selector dialog box, expand the Ledger tree, and expand the All Ledgers tree node.
- 18. Click the Vision Operations (USA) tree item and click **Select** (right arrow) to move it to the Selections panel.
- **19.** Click **OK**.
- 20. Repeat the preceding steps to complete the following dimension fields:

Field	Value
Company	"All Company Values","00"
Sub-Account	"All Sub-Account Values", "0000"
Product	"All Product Values", "[Product]@[000]"



Field	Value
Currency	"USD"

- 21. From the Rule Palette, drag Allocation to the Rule Designer panel.
- 22. In the Allocate Wizard, click in the Value column of the AccountingPeriod field.
- **23.** Click the **Actions** button to the right of the field.
- **24.** Select Variable.
- 25. In the Select Variable dialog box, select Database in the Category field.
- **26.** Click the **Balancing_Segment_Value** field, and click **OK**.
- Click Next.
- **28.** Click in the Select Value column of the **Company** field.
- **29.** Click the **Actions** button to the right of the field.
- **30.** Select Member.
- 31. In the Member Selector dialog box, expand the Company tree.
- 32. Expand the All Company Values tree item.
- 33. Click the 00 tree item.
- **34.** Click **Select** (right arrow) to move it to the Selections panel.
- 35. Click Next.
- **36.** Use the preceding procedure to complete the following Dimension fields:

Field	Description	Action
Department	"[Department]@[000]"	Member
Account	an account	Member

- 37. Click Save and then Exit.
- 38. Click Save and then OK.

Creating an Allocation Rule with a Formula Component: Worked Example

This example demonstrates how to create an allocation rule with a formula component.

You are the General Accountant for InFusion America Inc. You are tasked with creating an allocation rule with a formula component.

Rules are based on formulas. The formulas use multiple criteria. For example, you can use account balances or statistical amounts to allocate shared revenue or costs across multiple organizational units. You can define complex computations based on variables from various charts of accounts. You can group journal formulas together and execute them sequentially to update account balances in a step-by-step process.

Creating an Allocation Rule with a Formula Component

- 1. Click the Navigator.
- 2. Click the Journals link under General Accounting to access the Journals work area.
- 3. Click the Create Allocation Rules link.
- 4. In the Enterprise Performance Management System Workspace, click the **Navigate** menu.



- 5. Select Administer and then Calculation Manager.
- 6. On the System View tab of Calculation Manager, expand the Essbase tree.
- 7. Expand the VF_USA_Accounting_Flexfield tree item.
- 8. Expand the db tree node.
- 9. Right-click the Rules tree item and select New.
- 10. In the New Rule dialog box, enter EMEA Overhead Distribution in the Name field and click OK. You design a rule by dragging components from the Rule Palette to the Rule Designer flow chart. Each component performs a separate task. You can add Point of View, Allocation, or Formula components to your rule. You enter additional rule details on the Properties tab.
- 11. On the Properties tab, complete the following fields:

Field	Description
Name	Reserve for bad debts.
Description	This formula rule calculates the reserve needed for bad debts.

- 12. In the Rule Palette area, drag the Point of View object to the Rule Designer area.
- **13.** In the Point of View area, enter POV in the **Caption** field.
- **14.** Click in the **Ledger** field.
- **15.** Click the **Actions** button to the right of the **Ledger** field.
- **16.** Select Member.
- 17. In the left panel of the Member Selector dialog box, expand the Ledger tree.
- **18.** Expand the All Ledgers tree item.
- 19. Click the EMEA PC PL tree item.
- 20. Click the **Select** (right arrow) button to move the ledger to the right panel.
- **21.** Click **OK**.
- **22.** Repeat the preceding steps to complete the following fields:

Field	Description
Company	"[All Company Values]. [3888]"
Cost_Center	"000"
Program	"[All Program Values]. [0000]"
Location	"[Location]@[0000]"
Division	"[All Division Values]. [0000]"
Product	"[All Product Values]. [0000]"
Intercompany	"[All Intercompany Values]. [0000]"
Currency	"USD"
Currency Type	"Total"



Field Description

- 23. Click the Actions button to the right of the AccountingPeriod field.
- **24.** Select Variable.
- 25. In the Select Variable dialog box, select Database in the Category field.
- **26.** Select the Accounting Period variable and click **OK**.
- 27. To define the formula, drag the Formula object from the Rule Palette to the POV loop in the Rule Designer.
- 28. On the Formula tab, enter Bad Debts Calculation in the Caption field.
- 29. Click in the Enter formula field.
- **30.** Click the **Actions** button to the right of the **Enter formula** field.
- **31.** Select Member.
- 32. In the Member Selector dialog box, select Scenario in the **Dimensions** field.
- 33. Expand the Scenario tree and click the Actual tree node.
- **34.** Click the **Select** (right arrow) button to move Actual to the Selections panel.
- 35. Repeat the preceding steps to select the remaining dimensions and members for the formula.

Dimension	Member
Balance Amount	Ending Balance
Amount Type	YTD
Account	1399

- 36. Click OK.
- **37.** To enter a formula to create 5% of total accounts receivable as reserve for bad debts, in the **Enter formula** field, at the end of the formula you just created, enter *.05*-1.
- **38.** To specify the target members, click in the **Enter member name** field.
- 39. Click the **Actions** button.
- **40.** Select Member.
- **41.** In the Member Selector dialog box, expand the Account tree until the account 75555 appears.
- **42.** Click account 75555.
- **43.** Click the **Select** button to move the account to the Selections panel.
- 44. Click **OK**.
- 45. To specify offset members, on the Formula tab, enter "13005" in the Offset Member field.
- 46. Click Save.
- 47. Click **OK**.
- **48.** Click the **Validate** button and then the **OK** button to acknowledge validation of the rule.
- 49. To deploy the rule, click the Quick Deploy button in the tool bar.
- 50. When the Deployment Status dialog box appears, click **OK** to acknowledge successful deployment.

Designing an Allocation Rule: Example

An allocation rule is a Calculation Manager object that consists of calculations. The calculations are grouped into components. A rule can contain one or more components.

You create an allocation rule for an Essbase application. The rule is represented graphically in a flow chart into which you can drag and drop components to design the rule.



Scenario

To design an allocation rule:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, do one of these tasks:
 - Select File menu, New, Rule. Expand the Essbase Application Type, the Application, and the Calculation Type, Plan Type, or Database.
 - Right click Rules and select New.
- **4.** In **New Rule**, enter the rule's name, the **Application Type** Essbase, and the **Application Name**. The application name must be a valid Essbase application.
- 5. Select the **Database**.
- 6. OK.
 - Note: If you right click **Rules** and select **New** to create a new allocation rule, the **New Rule** dialog is populated with the **Application Type**, the **Application**, and the **Calculation**, **Plan Type**, or **Database** you are working in within the System View.
- 7. To design the allocation rule, from the **Rule Palette**, drag new and existing objects, and drop them into the flow chart within the **Rule Designer**.
 - Note: You can also create new objects such as formulas and scripts independently of the rule, and add them to the rule later.
- **8.** From **New Objects**, drag and drop these components to insert a new component into the rule's flow chart:
 - Point of Views:
 - Allocations:
 - o Formulas:
- **9.** On **Properties**, enter properties for the rule.
 - Note: The number and contents of the tabs change as you add components to the rule and move along the rule's components in the flow chart. To enter properties and other information for the rule's components, select the component in the flow chart. You can optionally:
 - Edit the name by entering a new one of up to 50 characters. The name defaults from the New Rule dialog.
 - o Enter a description of up to 255 characters for the rule.
 - o Enter a caption for the rule. The caption displays below the rule's icon in the flow chart.
 - o Enter comments for the rule. For example, you may want to tell the users how to use the rule.
- **10.** For Essbase: On **Global Range**, specify what dimensions are common to all of the components in the rule by selecting values, for example, members, variables, and functions for each dimension. The values you select for the dimensions are the values that are calculated when the rule is launched.
 - a. Select values for a dimension by clicking its row in the **Select Value** column.



- b. When the Actions icon is displayed, click the icon, and select one of these:
 - Variable
 - Member
 - Function
- 11. For Essbase: On **Variables**, select **Merge Variables** to merge all instances of the same variable used in the allocation rule so only the first instance of each variable is displayed when the rule is launched. If you do not select this check box, all instances of each variable are displayed.
 - Note: If you select **Merge Variables,** the first value that the user enters for the runtime prompt is used for all subsequent occurrences of that runtime prompt during validation and launch.
- 12. On the **Usages** tab, you view which allocation rules and rule sets use the rule, if any. You cannot edit any of the information on this tab. The following information is displayed for the allocation rules and rule sets using the allocation rule:
 - Names
 - Application Name
 - Deployment Status
 - Validation Status
 - Description
 - Note: By default, an allocation rule is not used by any allocation rules or rule sets when create.
- 13. Repeat these steps for each component you want to add to the allocation rule.
 - Note: As you add components to an allocation rule, you can increase or decrease the size of the component icons and the amount of detail that is displayed in the flow chart. You can use the zoom bar to zoom in and out within the flow chart. You can select a component to view its properties and edit a component on the Properties tab.
- 14. Select File, Save.
- 15. After you design and save the rule, you can do any of these tasks:
 - Print it.
 - Validate it.
 - Deploy it.
 - Generate it from within Oracle General Ledger by clicking on: Navigator > General Accounting: Journals >
 Generate Allocations link.

Editing Allocation Rules: Example

You can edit the structure of an allocation rule by adding to, removing, or changing its components. You can also edit the properties of the allocation rule itself. You can edit these properties of an allocation rule:

· Name and caption



- Description and comments
- Range of dimensions and members
- Variables, you include in the allocation rule

Scenario

To edit an allocation rule:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the System View, expand the Essbase Application Type, the Application, or Database, and Rules. Do one of these tasks:
 - a. Right click the rule you want to edit, and select **Open**.
 - **b.** Select the rule you want to edit, and select **File**, **Open**.
- **4.** To edit the rule, in the **Rule Designer**, add new components, and copy and delete existing components, from the rule's flow chart.
 - Note: As you edit components in an allocation rule, you can increase or decrease the size of the component icons and the amount of detail that is displayed in the flow chart. To edit, you use the zoom bar to zoom in and out within the flow chart. When the flow chart is displayed in small or very small sizes, the component captions do not display, but you can place your mouse pointer over any icon to read its caption. Regardless of the size of the components in the flow chart, you can select a component to view its properties on the Properties tab.
- 5. To delete a component from the flow chart, select the component, right click it, and select **Remove**.
- **6.** To copy and paste a component, select the component, right click it, and select **Copy**. Then paste it into the flow chart.
- 7. To add a new component:
- 8. From **New Objects**, drag and drop components to insert a new component into the rule's flow chart:
 - Point of Views
 - Allocations
 - Formulas
- 9. From Existing Objects, drag existing objects from Essbase applications and drop them into the rule's flow chart.
- **10.** For Essbase: On **Global Range**, you can edit the values that is, members, variables, and functions that define the range of values to be calculated when the rule is launched.
- 11. Select values for a dimension by clicking its row in the **Select Value** column.
- **12.** When the Actions icon is displayed, click it, and select one of these:
 - Variable
 - Member
 - Member
- 13. For Essbase: On Variables, you can create variables for the rule.
- **14.** On **Usages** tab, you can view which rules and rule sets use the rule, if any. This is the information you can view about the rules and rule sets that use the rule:



- Note: On the **Usages** tab, you view which allocation rules and rule sets use the rule, if any. You cannot edit any of the information on this tab. The following information is displayed for the allocation rules and rule sets using the allocation rule:
- Names
- Database
- Application Name
- Deployment Status
- Validation Status
- Description
- 15. Select File, Save.

Printing Allocation Rules: Example

You can print an allocation rule's properties, its flow chart, and the details of its components. For example, if you print an allocation rule that contains a formula component for allocation expenses and shows:

- The formula syntax.
- The functions and variables that comprise the formula.
- A summary of the steps in the rule's flow chart but not in graphical form.
- The rule's properties.
- Note: You cannot print allocation rule sets.

Scenario

To print an allocation rule:

- 1. Navigator > General Accounting > Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, expand the **Essbase Application Type**, the **Application**, or **Database**, and **Rules**. Select the rule you want to print.
- 4. Select File, Print.
 - Note: You can also select File, Print from within the Rule Designer to print a rule.
- 5. In **Print Preview**, do these tasks:
- 6. Select the Print options:
 - Paper size
 - Print orientation: portrait or landscape.
- 7. Select General Rule Information to print the rule's description and other details from the Properties tab. The details include the rule's name, the application, its owner, the date the rule was created, and the date the rule was last modified.



- 8. Select Flow Chart and Expanded or Collapsed, to print the flow chart, and you want to print it with the component details expanded or collapse.
- 9. Select the number of pages you want to print the components across (horizontally). Select the number of pages to print the components down (vertically).
- 10. Select the Page Order options:
 - Down, then across: The components in the flow chart print down (vertically, as rows do) on the number of pages you specified in the previous step. Then the components print across (horizontally, as columns do) on the number of pages you specified in the previous step.
 - Across, then down: The components in the flow chart print across (horizontally, as columns do) on the number of pages you specified in the previous step. Then the components print down (vertically, as rows do) on the number of pages you specified in the previous step.
- 11. Select the remaining Rule Information options:
 - Select Summary, to print a summary of the components in the flow chart.
 - Select Variable Section to print information about any variables used in the rule.
 - Select **Detail Section** to print detailed information about the components in the rule.
 - Select Page break before sections to create a page break between summary, variable, and detail sections.
 This option is selected by default.
 - Select Nested Rules to print rules contained in other rules.
- 12. Select Generate PDF.
- 13. A PDF file of the rule is opened in Adobe Acrobat.
- **14.** Click the **Print** icon in Adobe Acrobat.
- 15. In the Print dialog, select the print options specific to the printer you are using, and click Print.

FAQs for Define Allocation Rules

How can I open an allocation rule?

You open an allocation rule from the **System View** that is displayed by default when you open Calculation Manager. You can also open a rule using **File**, Open from within the tab of another rule, rule set, component, or template.

To open an allocation rule:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the System View, expand the Essbase Application Type, the Application, or Database, and Rules.
- 4. Do one of these tasks:
 - o Right-click the rule you want to open, and select **Open**.
 - Select the rule you want to open, and select File, Open.

How can I save an allocation rule?

You must save an allocation rule after you create or edit it. When you save the allocation rule, it is saved to the application and application type for which you created it. After you save it, you can validate and deploy it in Calculation Manager. You can generate it in Oracle Fusion General Ledger.



To save an allocation rule after you create or edit it, select File, Save.

Note: To see the allocation rule in the **System View** after you save it, you may refresh the application list. To do this, right-click the application type, the application, or the database (for Essbase), and select **Refresh**.

How can I save an allocation rule with a different name?

You can save an allocation rule with a different name using **Save As**. You can also copy a rule from one rule set to another within the same rule set type using **Save As**. **Save As** creates a copy of the original rule with a different name to distinguish it from the original.

To save an allocation rule with a different name:

- In the System View, expand the Essbase Application Type, the Application, or Database, and Rules.
- 2. Right-click the rule you want to save with a different name, and select Open.
- 3. In the Rule Designer, select File, Save As.
- 4. In Save As, enter the rule's new name, and select the Application Name.
- 5. Select the **Database**.
 - Note: You cannot change the database of a rule you save with a different name.
- 6. Click OK. The new rule is added to the application list in the System View.

How can I delete an allocation rule?

You delete an allocation rule from the **System View**. You can delete an allocation rule only if it is not used by other rules or rule sets. If the rule is being used, you must remove the allocation rule from the rules and rule sets using it, or make copies of it for the rules and rule sets using it, before you delete it. To see if a rule is used by other rules and rule sets, you can show the usages of the rule.

To delete an allocation rule:

- 1. In the System View, expand the Essbase Application Type, the Application, or Database, and Rules.
- 2. Make sure the rule you want to delete is not being used by another rule set or rule.
- 3. Right click the rule you want to delete, and select **Delete**.
- 4. Click **OK** to confirm deletion of the rule.

Define Allocation Rule Sets

Allocation Rule Sets: Explained

You create an allocation rule set by combining allocation rules or allocation rule sets that can be generated sequentially. You add rules and rule sets to a rule set by dragging and dropping them into it.

After you create and save the rule set, you can validate and deploy it. Then you can generate it in Oracle General Ledger.



Note: Rule sets are supported in Essbase aggregate storage applications used in Oracle General Ledger in sequential mode only.

Creating an Allocation Rule Set: Example

You can create an allocation rule set from the **System View**. You can also create an allocation rule from the **List, Custom**, and **Deployment** views and from within the **Ruleset Designer**.

Scenario

To create an allocation rule set:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. Enter the rule set's Name.
- 4. Enter the Application Type: Essbase).
- 5. Select an Application Name.
- Select a Database.
 - Note: From the **System View**, if you right click **RuleSets** and select **New** to create a new allocation rule set, the **New Ruleset** dialog is populated with the application type, the application, and the database in which you are working.
- 7. Click **OK**. The new rule set is displayed in the **Ruleset Designer**.

Creating an Allocation Rule Set With Deployment: Worked Example

This example demonstrates how to create an allocation rule set.

You are the General Accountant for InFusion America Inc. You are tasked with creating an allocation rule set.

You create rule sets by combining two or more related rules (or rule sets) so you can launch the rules or rule sets sequentially.

Creating an Allocation Rule Set

- 1. Click the Navigator.
- 2. Click the **Journals** link under General Accounting to access the Journals work area.
- 3. Click the Create Allocation Rules link.
- 4. In the Enterprise Performance Management System Workspace, click the Navigate menu.
- 5. Select Administer and then Calculation Manager.
- 6. On the System View tab of Calculation Manager, expand the Essbase tree.
- 7. Expand the VF_USA_Accounting_Flexfield tree item.
- 8. Right-click the Rules tree node and select New.
- 9. In the New RuleSet dialog box, enter EMEA Overhead Distribution, and click OK.
- 10. Expand the VF_USA_Accounting_Flexfie tree in the Rule Set Palette, then expand the db tree node and then the Rules tree node.
- 11. Drag EMEA Overhead Distribution from the Rule Set Palette to the Ruleset Designer panel.
- 12. Drag EMEA Allocation of Indirect Costs from the Rule Set Palette to the Ruleset Designer panel.
- 13. Click EMEA Overhead Distribution Rule Set in the Ruleset Designer.



- 14. Click the Variables tab in the lower panel of the work area.
- 15. Select Merge Variables.

If a rule set has the same variable used across rules in a rule set, you can merge the rules. Then at generation time, the input is taken from that variable as the run time prompt.

16. Click Save and then OK.

Before you deploy rule sets, you validate them to ensure that there are no syntax errors. Validation ensures that all of the following criteria are met:

- o All members of all dimensions in the application are valid.
- o All functions exist, have the correct number of parameters, and are valid for the application type.
- All variable references that are used in rules are valid. Replacement variables are replaced with the correct strings and then validated.
- The generated script contains no syntax errors.

You can deploy rules and rule sets to Oracle Fusion General Ledger. You execute a partial deployment by deploying one or more (but not all) rules and rule sets. You execute a full deployment by deploying all rules and rule sets in the current application.

Rules and rule sets that are deployed can be generated from the Allocation Generation page.

- To deploy multiple rules and rule sets but not all; in Deployment View, select the check boxes of the rules and rule sets that you want to deploy.
- To deploy only one rule or rule set in the current application; then in Deployment View, select the one rule or rule set option.

Designing an Allocation Rule Set: Example

After you create a rule set in the **New RuleSet** dialog, the rule set is displayed in the **Ruleset Designer**.

Scenario

To create an allocation rule set:

- Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, expand the Essbase application type and the application.
- 4. For Essbase: Right click RuleSets and select New.
 - Note: For Essbase applications, only one rule sets node for each application at the same level as the databases.
- 5. In New RuleSet, do these tasks:
 - a. Enter the rule set's name
 - **b.** Select the Application Type as **Essbase**
 - c. Select the Application Name
 - d. As you selected Essbase as the application type, select the **Database**.
 - e. OK.



- In the Ruleset Designer, to create the rule set, from Ruleset Palette, drag existing rules and rule sets and drop them into the flow chart.
 - Note: You can use the up and down arrow buttons below the **Navigate** menu to reorder the rules in the rule set. To move a rule up or down, select the rule and click the up or down arrow button until the rule is in the correct location. Rules in General Ledger applications are launched sequentially within a rule set, so the order of the rules is important.
- 7. On **Properties**, enter properties for the rule set. In the **Ruleset Designer**, if you select a rule or rule set within the rule set you are creating; its properties are displayed on Properties instead of the new rule set's properties.

Optionally, enter the following:

- a. The name by entering a new one of up to 50 characters. The name defaults from the **New RuleSet** dialog.
- **b.** A description of up to 255 characters for the rule set.
- c. Comments for the rule set. For example, you may want to enter a comment that describes what the allocation rule set does.
- 8. On **Usages** tab, you view which allocation rules and rule sets use the rule, if any. You cannot edit any of the information on this tab. The following information is displayed for the allocation rules and rule sets using the allocation rule:
 - a. Names
 - **b.** Database
 - c. Application Name
 - d. Deployment Status
 - e. Validation Status
 - f. Description
 - Note: By default, a rule set is used by no other rule sets when you create it.
- 9. On Variables, select Merge Variables to merge all instances of the same variable used in the rules within this rule set so only the first instance of each variable is displayed when the rule is launched. If you do not select this check box, all instances of each variable are displayed.
 - Note: If you select **Merge Variables**, the first value that the user enters for the runtime prompt is used for all subsequent occurrences of that runtime prompt during validation and launch.
- 10. Select File. Save.

Adding an Allocation Rule to an Allocation Rule Set: Examples

You can add an allocation rule to an allocation rule set that belongs to the same application type. The rules in the rule set can be launched sequentially or simultaneously.

Scenario

To add an allocation rule to an allocation rule set:

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. Expand RuleSets, right click the rule set you want to open, and select Open.



- Note: Only one rule set node exists for each application at the same level as the plan types and databases.
- In the Ruleset Designer, in Existing Objects, expand the application and the plan type or calculation type that contains the rule you want to add.
- 4. To add the rule, drag and drop it into the Ruleset Designer.
- 5. Repeat step 4 for each rule you want to add to the rule set.
- 6. Select File. Save.

Editing Allocation Rule Sets: Examples

You can edit the following properties of an allocation rule set:

- Allocation rule components
- Allocation rule name
- Allocation rule description
- Allocation rule comments

Scenario

To edit an allocation rule set:

- 1. Navigator > General Accounting > Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the System View, expand Essbase Application Type and the Application.
- 4. Expand your rule set by right-clicking the rule set you want to edit, and selecting **Open**.
 - Note: Only one **rule set** node exists for each application at the same level as the plan types and databases.
- 5. In the **Ruleset Designer**, add, copy, delete and change the order of new rules and rule sets:
 - a. To delete a rule or rule set from the rule set, select the rule or rule set, right-click it, and select **Remove**.
 - **b.** To add a rule or rule set to the rule set, from **Existing Objects**, drag existing rules and rule sets from Essbase applications, and drop them into the **Ruleset Designer**.
 - Note: The rules and rule sets you add to the rule set must belong to the same application type as the rule set you are editing.
 - c. To open a rule or rule set in the rule set, right-click the rule or rule set, and select **Open**.
 - **d.** To reorder the rules and rule sets within the rule set, use the up and down arrow buttons below the **Navigate** menu. To move a rule or rule set up or down, select it and click the up or down arrow button until in the correct location.
- **6.** On **Properties**, edit properties of the rule set. (In the **Ruleset Designer**, if you select a rule that you added to this rule set, the properties are displayed on the Properties tab.)
- 7. Optional: Edit the name by entering a new one of up to 50 characters. (The name defaults from the New RuleSet dialog.)
- 8. Optional: Edit the description by entering a new one of up to 255 characters.



- 9. Edit the **Enable Parallel Execution** selection. If you want the rules and rule sets in the rule set to run simultaneously, select this option, to run sequentially, clear this option. By default, the rules and rule sets in a rule set run sequentially: each rule or rule set in the rule set must run without errors before the next rule or rule set is run.
- 10. If the rule set contains nested rule sets which have a different Enable Parallel Execution setting than the parent rule set, the setting of the nested rule set applies. For example, if you have rule set1 that is marked for parallel processing and it contains rule1, rule2, and rule set2 that is marked for sequential processing, the rules and rule sets in rule set2 are processed sequentially, even though rule set1 is marked for parallel processing.
- 11. Edit the comments.
- 12. On the **Usages** tab, you view which allocation rules and rule sets use the rule, if any. You cannot edit any of the information **that appears on this** tab. The **following** information is displayed for the allocation rules and rule sets using the allocation rule:
 - a. Names
 - b. Calculation or Plan Type
 - c. Application Name
 - d. Deployment Status
 - e. Validation Status
 - f. Description
- **13.** Select **File**, **Save**.

Copying an Allocation Rule Set to Another Application: Example

From the **System View**, you can copy an allocation **rule set** to another application of the same application type (Essbase) or database.

Note: Allocation rule sets are not supported in Essbase Aggregate Storage or Block Storage applications, other than Aggregate Storage applications used in Oracle General Ledger.

Scenario

Use the following steps to copy a rule set to another application:

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. Expand RuleSets.
 - Note: Only one rule set node exists for each application at the same level as the plan types and databases.
- 3. Right click the allocation rule set you want to copy, and select **Copy To**.
 - Note: You can also copy an allocation rule set when you are working with it in the **Ruleset Designer**, and paste it into another allocation rule set or allocation rule.
- 4. Select the rule set name, select **Edit**, **Copy**, open the rule or rule set into which you want to copy it, and select Edit, Paste.
- 5. In **Save As**, enter a new name for the allocation rule set, or accept the default name, and select an application and calculation or plan type. Click **OK**.



Note: You cannot copy the allocation rule set to more than one application and calculation or plan type.

The new allocation rule set is added to the application and calculation or plan type you selected. To see it in the **System View**, you must refresh the application list. To refresh the application list, click the **Refresh** icon on the toolbar. You can also refresh rule sets or any level above it in the application list to see the new rule set.

Saving an Allocation Rule Set: Example

You must save an allocation rule set after you create or edit it. When you save the allocation rule set, it is saved to the application and application type for which you created it. After you save it, you can deploy, validate, and generate it. You can deploy and validate it in Calculation Manager; you can generate it from Oracle General Ledger.

Scenario

To save an allocation rule set after you create or edit it, select **File**, **Save**, or click the **Save** icon.

Note: To see the allocation rule set within the **System View** after you save it, you must refresh the application list. To do this, right click the application type, the application, the database (Essbase), and select **Refresh**. You can also click the **Refresh** icon on the toolbar to refresh the application list in the **System View**.

Saving an Allocation Rule Set with a Different Name: Example

You can save an allocation rule set with a different name using **Save As**. Saving it with a different name creates a copy of the rule set.

Scenario

- 1. In the System View, expand the Essbase Application Type and the Application.
- Expand RuleSets.
 - Note: For Essbase applications, only one rule set node exists for each application at the same level as the plan types and databases.
- 3. Right-click the rule set you want to save with a different name, and select **Open**.
- 4. In the Ruleset Designer, select File, Save As.
- 5. In Save As, enter the rule set's new name, and select an application. Click OK.
 - Note: You cannot change the application type of a rule set you save with a different name. The new rule set must have the same application type as the rule set from which the rule set is created.

The new rule set is added to the application list in the **System View**.



Deleting an Allocation Rule Set: Example

You delete an allocation rule set from the **System View**. You can delete an allocation rule set only if it is not being used by other allocation rule sets. To see if it is being used by other rule sets, you can show its usages. If it is being used, you must remove it from the allocation rule sets that are using it, or make copies of it for the allocation rule sets that are using it, before you delete it.

Scenario

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. Expand RuleSets.
- 3. To make sure the rule set is not being used by another rule set, right click it, and select **Show Usages**.
- 4. Right click the rule set you want to delete, and select **Delete**.
- 5. Click **OK** to confirm deletion of the rule set.

Refreshing Allocation Rules or Allocation Rule Sets: Example

In the **System View**, you can refresh any level of the application list. You can refresh:

- Application Type
- Application
- Database
- One or Multiple Rule Sets or Rules

By default, when you refresh an application, application type, or database, all of the rules, rule sets, components, and templates belonging to it are refreshed.

However, refreshing the rule sets or rules within an application does not refresh higher levels in the application list or rule sets or rules that belong to other applications.

Note: You can also click the **Refresh** icon on the toolbar to refresh the entire application list in the **System** View.

Scenario

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. To refresh rule sets, right click **RuleSets**, and select **Refresh** or to refresh rules, expand the database, right click **Rules**, and select **Refresh**.
 - Note: You can also right click the application type, the application, or database that contains the allocation rules you want to refresh, and select **Refresh**.



Showing the Usages of a Rule or Allocation Rule Set: Example

You can display the allocation rules and rule sets that are using a rule or allocation rule set. Viewing the usages of rules or rule sets is useful when deleting rules or rule sets and you must know what objects are using it.

Scenario

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. To show the usages of a rule set, expand **RuleSets**.
- 3. Right-click the rule set whose usages you want to see.
- 4. Select **Show Usages** or to show the usages of a rule, expand the database, and **Rules**.
- 5. Right-click the rule whose usages you want to see, and select Show Usages.
- 6. You can view this information about the rule or allocation rule set:
 - a. Names
 - **b.** Database
 - c. Application Name
 - d. Deployment Status
 - e. Validation Status
 - f. Description
 - Note: You can also view a rule or rule set's usages from within the Rule or Ruleset Designer on the Usages tab.
- 7. After you review the information, click **OK**.

Changing the Owner of an Object: Example

You can change the owner of an object such as a rule, rule set, or formula in the **System View**. The application to which it belongs must be deployed. By default, an object's owner is the user that creates it, unless the user changes the ownership. Users can edit only objects they own, with the exception of administrators who can edit any objects.

Scenario

- 1. In the System View, expand the Essbase Application Type and the Application.
- To change the ownership of a rule set, expand RuleSets or to change the ownership of a rule, expand the database, and then expand Rules.
- **3.** Right-click the object, and select **Change Ownership**.
- 4. In Change Owner, select the owner to whom you want to transfer ownership of the object.
- 5. Click OK.
 - Note: The user you assigned ownership to can edit the object.



FAQs for Define Allocation Rule Sets

How can I open an allocation rule set?

You open an allocation rule set from within the **System View**. You can also open a rule set from within the **Ruleset Designer**, by selecting **File**, then **Open**.

To open an allocation rule set:

- 1. In the **System View**, expand the Essbase application type and the application.
- 2. For Essbase: Expand RuleSets, right-click the rule set you want to open, and select Open.
 - Note: For Essbase applications, there is only one rule set node for each application at the same level as the databases.

How can I open an allocation rule within an allocation rule set?

You can open an allocation rule from within an allocation rule set from the **System View** or from the **Ruleset Designer**. To open an allocation rule within an allocation rule set:

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. Expand your rule set that contains the rule you want to open.
 - Note: For Essbase applications, only one rule set node exists for each application at the same level as the databases.
- 3. Right-click the rule you want to open, and select **Open**.
 - Note: You can also open a rule that belongs to an allocation rule set from within the **Ruleset Designer**. To do this, in the **Ruleset Designer**, right-click the rule, and select Open.

How can I remove an allocation rule from an allocation rule set?

When you remove a rule from an allocation rule set, the rule is not deleted. The rule exists independently of the rule set in the database.

To remove an allocation rule from an allocation rule set:

- 1. In the System View, expand the Essbase Application Type and the Application.
- 2. Expand RuleSets, right-click the rule set you want to open, and select Open.
 - Note: Only one **rule sets** node exists for each application at the same level as the plan types and databases.



3. In Ruleset Designer, right-click the rule you want to remove, and select Remove.

Select File, Save.

Working with Point of View and Allocation Components

Working with Point of View Components: Overview

Every data value in a report is derived from the intersection of a member from each dimension in the Essbase database connection. Oracle Fusion Financial Reporting enables a designer to place these dimensions on the report grid or user point of view (POV). Report viewers can change the member selected for dimensions on the user POV. This enables report viewers to customize the reports to fit their needs. The user POV can also be used in books.

You create and edit POV components from within a rule to set members, variables, and functions that comprise the global range of the POV component. You nest a POV of View component within another POV component.

- Allocation components contain calculations for distributing data from members at one level in the database outline to other members in the outline.
- Formula components contain calculation statements that you design using members, functions, and variables.

Creating a Point of View Component: Example

You create a Point of View (POV) component from within a rule to set members, variables, and functions that comprise the global range of the POV component.

You can also define or edit the caption that displays above the component in a flow chart and the comments that are entered for the values selected for each of the dimensions in the POV.

Scenario

To create a Point of View component:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, select **File** menu, **New, Rule**. Expand the Essbase Application Type, the Application, and the Calculation Type, Plan Type, or Database and **Rules**.
- 4. Right click rules you want to open, and select Open. The rule is displayed in the Rule Designer.
- 5. After you determine where in the rule's flow chart you want to create the POV component, from the New Objects Palette, drag the POV component and drop it into that location in the flow chart. The POV object is displayed as two circles with arrows inside them.
- 6. On the **Point of View** tab, enter a caption to identify the POV component. The caption is displayed above the component in the flow chart of any rule that uses it.
- 7. Optional: Do one of these tasks to define the POV global range:
 - a. Click Variable Selector to select or create variables to define the POV. If you select a variable, you can select Link Variable Dynamically to ensure the variable is updated dynamically when changes are made to it.
 - **b.** Click **Member Selector** to select members to define the POV.
 - c. Click in the row of a dimension in the **Value** column to type the names of members that define the POV.



- d. Click in the row of a dimension, click the **Actions** icon, and select one of these options to enter members:
 - Members
 - Variables: You can use a variable to fill the POV component. The variable must be defined at the database level and must be of the Member Range type.
 - Functions: The functions you enter should return level 0 members only and should include a @ symbol before the function name. You can enter these functions:
 - @LevelODescendant
 - @Sibling
 - @UDA
 - @Attribute
 - Note: The LevelODescendant and Sibling functions require a member name as a parameter.
- Note: If a global range is defined for the rule for which you are creating the POV component, the Point of View tab displays the rule's member selections by default. To see if a global range is defined for the rule, select the Begin or End tab in the flow chart. Then click on the Global Range tab to see if any members, functions, or variables are defined.
- 8. If you want to enter a comment for the members you select for a dimension, click Comment.
- 9. Click **Reset Grid** to clear any entries you made to the grid.
- 10. Select File, Save.

Editing a Point of View Component: Example

You can edit the members, variables, and functions that comprise the global range of the Point of View (POV) component. You can also edit the caption that displays above the component in a flow chart and the comments that are entered for the values selected for each of the dimensions in the POV.

Scenario

To edit a POV component:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, select **File** menu, **New, Rule**. Expand the Essbase Application Type, the Application, and the Calculation Type, Plan Type, or Database.
- 4. Right click the rule, and select **Open**.
- 5. In the **Rule Designer**, select the POV component you want to edit in the flow chart to display its properties. You can edit any of these properties of a POV component.
 - a. The caption that displays above the POV component in the rule's flow chart.
 - **b.** The members, variables, and functions that define the POV.
 - **c.** Whether any variables used in the POV component are updated dynamically when changes are made to the variables.
 - d. Whether comments are entered for the dimensions and members that define the global range of the POV.



- **e.** Whether the values of the members in the POV component are calculated when the rule to which it belongs is validated or launched.
- **6.** Select **File**, **Save**.

Working with Allocation Formula Components: Examples

An allocation formula component is composed of formula calculation statements. These examples show performing the following actions on formula components:

- Creating
- Opening
- Editing
- Deleting
- Copying and pasting
- Showing usages

Creating Formula Components

To create the calculation statements of a formula, you enter or select members and variables. As you create the formula, each of its calculation statements is listed in a row within a grid in the Component Designer. You can create a formula component from the **System View** or from within the **Rule Designer**. A formula component exists as an independent object in the database, so it can be shared among allocation rules.

- From the System View, expand the Essbase Application Type, the Aggregate Storage Application, and the Database.
- 2. Right-click Formulas, and select New.
- 3. Enter the formula's Name.
- **4.** Enter the **Application Type** (Essbase).
- 5. Select an **Application Name** such as Vision Chart of Accounts. The application name must be a valid Essbase aggregate storage application.
- 6. Select the Essbase Database.
 - Note: Right-click Formulas, and select New to create a new formula, the New Formula dialog. The dialog is populated with the application type, the application, and the database you are working in within the System View.
- 7. Click **OK**. The formula is displayed in the **Component Designer**.
- 8. In the Component Designer, on Properties tab, optionally complete these steps:
 - a. Verify that the **Shared** check box is selected by default.
 - Note: You cannot select or clear the **Shared** check box. To create a formula that is not shared, open an allocation rule, and then drag a new formula component into the allocation rule's flow chart. The shared check box is not selected. If you decide to make the formula shared, select the **Shared** check box.
 - **b.** Edit the formula's name by entering a new one of up to 50 characters. The name defaults from the **New Formula** dialog.



- Note: Names of members in Essbase aggregate storage applications can be the same. If there is a name that is not unique, the name is marked when the object is validated. If this occurs, you must enter the full path for the member. The full path syntax is [Dimension name]. [Parent name]. [Member name.]
- c. Enter a description of up to 255 characters for the formula.
- **d.** Enter a caption for the formula. The caption is displayed below the formula in the **Rule Designer** flow chart. If you do not enter a caption, the component's name is displayed in the flow chart.
- e. Enter comments for the formula. For example, you may want to tell users how the formula component should be used.
- 9. On **Formula**, enter a caption for the formula.
- 10. The **Offset Member** is the total amount of all of the formulas in the formula component. Click the **Ellipsis** icon to select a member.
 - Note: You can define an offset member manually within the formulas you create below. The offset defined in the formula component is calculated as the sum of all calculated amounts.
- **11.** To create a formula statement, click in the first formula statement row to the left of the equal sign. Then enter a member or cross dimension member selection, or click **Actions** to select:
 - A variable
 - One or more members. To type a cross dimension selection of members, enter each member name, separated by a right arrow. For example, mem1->mem2->mem3.
 - 0
- **12.** To complete the formula statement, click in the row to the right of the equal sign. Then enter a member or cross dimension member selection, or click **Actions** to select a variable or member.
- 13. For each formula statement row, click the Comments icon to enter optional comments about the formula statement.
- **14.** On Usages, you can view the rules that use the formula component.
- **15.** Select **File** > **Save**.

Opening Formula Components

You can open a formula component from the **System View** or within the **Rule Designer** flow chart of a rule that uses the formula component.

- From the System View, expand the Essbase Application Type, the Aggregate Storage Application, and the Database.
- 2. Expand Formulas.
 - Right-click the formula you want to open, and select Open.
 - Double-click the formula you want to open.
 - Note: The formula component opens in the **Component Designer**. To open a formula component within a rule, open it from within the rule's flow chart by right-clicking the formula component. Select **Open** or double-click the component.



Editing Formula Components

You can edit the formula statements that comprise a formula component and the formula component's comments, caption, name, and description.

- 1. From the System View, expand the Essbase Application Type, the Aggregate Storage Application, and the Database and the Formulas.
- 2. Select the formula component you want to edit.
- 3. Right-click the formula component, and select Open.
- 4. In the Component Designer, you can edit any of these properties of a formula component:
 - Caption
 - Formula statements
 - Name
 - Description
 - Comments
- 5. Select File, Save.

Deleting Formula Components

You can delete a formula component only if the component is not being used in any allocation rules. To see if any allocation rules are using the formula component, you can show the formula component's usages.

If the formula component is being used by a rule, and you don't want the formula component, remove it from the rule, then delete it. If the formula component is being used in a rule, and you no longer need the rule, you can delete the rule. If no allocation rules use the formula component, you can delete the component.

To delete a formula component:

- 1. In the System View, expand the Essbase application type, the Essbase aggregate storage application, the database, and **Formulas**.
- 2. Right-click the formula you want to delete, and select **Show Usages** to make sure no allocation rules are using the formula component.
- 3. Remove the formula component from any allocation rules that are using it.
- 4. In the System View, right-click the formula you want to delete, and select **Delete**.
- 5. Confirm deletion of the formula.

Copying and Pasting Formula Components

You can copy a formula component from a rule and paste it into the same, or a different, rule. You can also copy the contents of the grid within a formula component and paste the contents into the same, or a different, formula component. You cannot copy a formula component and paste it into another formula component or another component type.

To copy and paste a formula component:

- 1. In the System View, expand the Essbase application type, the Essbase aggregate storage application, the database, and **Rules**.
- 2. Right-click the rule that contains the formula component you want to copy, and select **Open**.
- 3. In the Rule Designer flow chart, right-click the formula component you want to copy, and select Copy.
- 4. Do one of these tasks:
 - To paste the formula component into the same rule flow chart, right-click in the location of the flow chart, and select Paste.



- To paste the formula component into a different rule flow chart, open the rule where you want to paste the component. Then right-click in the location, and select **Paste**.
- 5. Select File, Save.

Showing a Formula Component's Usages

You can see which allocation rules are using a formula component, and other information, by displaying the formula component's usages from the System View.

To show a formula component's usages:

- 1. In the System View, expand the Essbase application type, the Essbase aggregate storage application, the database, and **Formulas**.
- 2. Right-click the formula whose usages you want to see, and select Show Usages.
- **3.** You can view this information about the formula component:
 - The names of the allocation rules that are using the formula component.
 - o The application names of the allocation rules that are using the formula component.
 - o The database names of the allocation rules that are using the formula component.
 - The owner of the formula component.
 - Whether the allocation rules that are using the formula component are deployed.
 - Whether the allocation rules that are using the formula component are validated.
 - o A description of the allocation rules that are using the formula component.
- Note: You can also view a formula component's usages from within the Component Designer on the Usages tab.

Creating an Allocation Component: Example

You create an allocation component from within a rule; it exists only in that rule and cannot be shared among allocation rules.

Scenario

To create an allocation component:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, select **File** menu, **New, Rule**. Expand the Essbase Application Type, the Application, and the Calculation Type, Plan Type, or Database.
- 4. Right click the rule you want to open, and select **Open**. The rule is displayed in the Rule Designer.
- 5. After you determine where in the rule's flow chart you want to create the allocation component, from the **New Objects Palette**, drag the **Allocation** component and drop it into that location in the flow chart.
- 6. In the Calculation Manager, on the **Point of View** (POV) tab, for each dimension listed that you do not want to vary during the allocation, do one of these tasks, and then click **Next**.
 - a. Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values.
 - **b.** Click the **Member Selector** icon to select members and variables for each of the dimensions listed. Make sure that all members you select are valid level 0 members.
 - c. Select a dimension in the list, and click **Actions** to select a member or variable.



- Note: If you drop a POV component within another POV component, the second POV inherits the members, variables, and functions from the first (that is, upper) POV. In the **Member Selector**, the dimensions listed in the current step are available for selection from **Dimension**. This enables you to select members and functions for any of the dimensions listed in the current step.
- 7. In the Calculation Manager, on the **Source** for each dimension listed, select a member whose data you want to allocate by doing one of these tasks. You must select a member for each dimension listed. The source members can be non-level 0 members.
 - **a.** Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values. If the predefined selection does not enter a value for each dimension listed, you must enter a value for any dimensions that are empty.
 - **b.** Click the **Member Selector** icon to select a member for each of the dimensions listed.
 - **c.** Select a dimension in the list, and click **Actions** to select a member or variable. You cannot use functions in this step of the Allocation component.
 - **d. Optional**, to allocate a specific value, enter an amount to be allocated instead of the selections above.
- 8. If the source amount you want to allocate is zero, select one of these options from the drop-down list.
 - a. Select the next pool record.
 - **b.** Stop processing the allocation.
- 9. Click Next.
- 10. On **Allocation Range**, enter the parent member for the dimensions you want to use for the allocation. To enter the parent member, do one of these tasks, and then click **Next**.
 - a. Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values.
 - **b.** Click the **Member Selector** icon to select the parent member for the dimension to which to allocate the data.
 - **c.** Enter a parent member, or select a dimension in the list. Click the **Actions** icon to select the parent member (of the main dimension) to which to allocate the data. The data is allocated to the level 0 member (that is, the lowest member in the outline, with no members beneath it) below the parent member in the database outline.
- **11.** On **Target**, for the remaining dimensions, select a level 0 member to which to allocate the data. Perform one of these tasks and click **Next**.
 - a. Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values.
 - b. Click the **Member Selector** icon to select members for each of the dimensions listed.
 - c. Select a dimension in the list, and click the Actions icon to select a member or variable.
- **12.** On **Offset**, perform one of these tasks and click **Next**:
 - a. Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values.
 - **b.** Click the **Member Selector** icon to select members for each of the dimensions listed.
 - c. Select a dimension in the list, and click the **Actions** icon to select a member or variable.
 - Note: You must specify members for the offset; you cannot leave it empty.
- 13. Optional: On **Exclude**, select any members you want to exclude from the allocation. Perform one of these tasks and click **Next**.
 - a. Select a predefined selection from **Use Predefined Selection** to populate the dimensions listed with values.
 - b. Click the **Member Selector** icon to select members for each of the dimensions listed.
 - c. Select a dimension in the list, and click the **Actions** icon to select a member or variable.



14. On **Basis** perform these tasks:

- Select an allocation method to specify how the data should be allocated.
 - i. Select Allocate evenly to allocate data values in the allocation range evenly. Then on Basis Options for evenly method, specify what you want to be done if the basis is negative, zero, has missing values, or if all members are excluded.
 - ii. Select Allocate using a basis to calculate a percentage to be applied to each member in the allocation range. Then on Basis Options, specify what you want to be done if the basis is negative or equal to zero.
- **b.** Any dimension members you do not specify are inherited from the POV you defined previously, but you can override those POV selections by doing one of these tasks:
 - Select a predefined selection from Use Predefined Selection to populate the dimensions listed with values.
 - ii. Click the **Member Selector** icon to select a member for each of the dimensions listed.
 - iii. Select a dimension in the list, and click the **Actions** icon to select a member or variable.

15. Click Next.

- 16. On Rounding, complete these steps. The members you select in this step must be a part of the allocation range.
 - **a.** Enter the number of decimal places to use for this allocation, or click the **Actions** icon to select a member or variable that represents this value.
 - **b.** Select where to place the rounding difference.
 - i. Select **Define location** to specify a member or members on which to place the rounding difference. Perform the following steps.
 - Select a predefined selection from Use Predefined Selection to populate the dimensions listed with values.
 - b. Click the **Member Selector** icon to select a member for each of the dimensions listed.
 - c. Select a dimension in the list, and click the **Actions** icon to select a member or variable.
 - ii. Select **Use biggest value** to round data values to their largest values.
 - iii. Select Use smallest value to round data values to their smallest values.
 - iv. Select **Discard rounding error** to use allocated data values as they are.
- 17. Click Finish.

Editing an Allocation Component: Example

You can edit an allocation component by opening the rule to which it belongs. When the rule is displayed in the Rule Designer, you can view the allocation component's properties by selecting it in the rule's flow chart.

Scenario

To edit an allocation component:

- 1. Navigator > General Accounting: Journals > Create Allocation Rules link.
- 2. Navigate menu > Administration > Calculation Manager.
- 3. In the **System View**, select **File** menu, **New, Rule**. Expand the Essbase Application Type, the Application, and the Calculation Type, Plan Type, or Database and Rules.
- 4. Select the rule that contains the allocation component you want to edit.
- 5. Right click the rule, and select Open.



- **6.** In the **Rule Designer**, select the allocation component you want to edit in the flow chart to display its properties. You can edit any of these properties of an allocation component.
 - a. The member whose data you want to allocate.
 - **b.** The level 0 members to which you want to allocate data.
 - c. The data and the amount of the data you want to allocate.
 - d. How you want the data processed:
 - The total amount of the data allocated written to an offset member.
 - The data allocated evenly or in different amounts using a driver.
 - The allocated data rounded, and if so, how it should be rounded.
- 7. Select File, Save.

FAQs for Working with Point of View and Allocations Components

How can I open a point of view or an allocation component?

You open a point of view or allocation component from within the flow chart of the rule to which it belongs.

Note: You cannot open the allocation component from the System View.

To open a point of view or an allocation component:

- 1. In the System View, expand the Essbase Application Type, the Application, Database, and Rules.
- 2. Right-click the rule that contains the component you want to open, and select **Open**. The rule is displayed in the **Rule Designer**.
- 3. When the rule opens, click the point of view component or double-click the allocation component in the rules flow chart to open the component.

How can I save a point of view or allocation component?

You save a point of view or an allocation component when you save the rule to which it belongs in the **Rule Designer**. Unlike the formula components, point of view and allocation components cannot exist independently for the rule to which they belong.

To save a point of view or allocation component:

• Select **File** and **Save**, or click the **Save** icon, once you finish designing a point of view or an allocation component.

How can I copy and paste a point of view or allocation component?

You can copy a point of view or allocation component from within the rule in which it is used, and you can paste the component into the same or a different rule.



To copy and paste a point of view or allocation component in a rule flow chart:

- 1. In the System View, expand the Essbase Application Type, the Application, Database, and Rules.
- 2. Right click the rule that contains the point of view or allocation component you want to copy and paste, and select **Open**. The rule is displayed in the **Rule Designer**.
- 3. In the Rule Designer, right click the point of view or allocation component you want to copy in the rules flow chart, and select Copy to copy only the component or Copy Group to copy the component and the associated components within it.
- 4. Do one of the following tasks:
 - If you want to paste the component into the same rule, right click the location in the flow chart where you want to paste the component, and select **Paste**.
 - If you want to paste the component into a different rule, open the rule, right-click the location in the flow chart where you want to paste the component, and select **Paste**.
 - Note: You can also select Edit and Paste.
- 5. Select File and Save.

How can I delete a point of view or an allocation component?

You delete a point of view or an allocation component from the **System View**. Since point of view or allocation components can be used in only one rule, you can delete these components by removing them from the rule to which they belongs to.

To delete a point of view or an allocation component:

- 1. In the System View, expand the Essbase Application Type, the Application, or Database, and Rules.
- 2. Right click the rule that contains the point of view or allocation component you want to delete, and select **Open**. The rule is displayed in the **Rule Designer**.
- 3. In the Rule Designer, select the point of view or allocation component you want to delete in the flow chart.
- 4. Right click the point of view or allocation component.
- **5.** Select **Remove** to confirm the removal of the component.
 - Note: If the allocation component is within the point of view component, then removing the point of view component removes the allocation component.
- 6. Select File and Save.

Generate Allocations and Periodic Entries

Generating Allocations and Periodic Entries Manually: Worked Example

This example demonstrates how to generate an allocation or periodic entry manually from the Oracle Fusion General Ledger.

You are the General Accountant for InFusion America Inc. You have created allocation and periodic journal entry definitions for several monthly entries. You now generate these entries.



Note: Schedule allocations and periodic entries in the Journals work area for automatic generation. The accounting period automatically increments for each subsequent run if defined as a Run Time Prompt for the Allocation Rule selected.

Prior to generating the allocation and periodic entries, the following tasks must be completed:

- The period is set to Open or Future Enterable. You post in open periods, but generation can take place in either
 an open or future enterable period.
- The rules or rules sets have been defined, validated, and deployed successfully from the Calculation Manager.
- The journal balances, which are inputs for the allocation or periodic rules, are entered and posted in the proper period.

Generating Allocations and Periodic Entries Manually

- 1. From the Navigator, click the **Journals** link to open the Journals work area.
- 2. In the task pane of the Journals page, click the **Generate Allocations** link to open the Submission page.
- 3. Optionally select one or all of the following options:
 - Print Output
 - E-mail me the output
 - Notify me when this process ends
- 4. Select a rule or rule set from the list of values.
- 5. Enter the submission parameters, including **Ledger**, **Balancing Segment Value**, and **Period**. The application automatically sets the last day of the submission period as the Accounting Date and Calculation Effective Date.
- 6. Accept the selected check box for the **Post Allocations** option to enable the process to post the journal entries.

If you deselect the check box for the Post Allocations option, you must either manually post or define an AutoPost Criteria Set to post automatically.

7. Click Submit.

After the generation process is complete, the journal entries created by the process are available for inquiry on the Journals page.

Manage Revaluations

Revaluation Process: Explained

The revaluation process is used to adjust account balances denominated in a foreign currency. Revaluation adjustments represent the difference in account balances due to changes in conversion rates between the date of the original journal and the revaluation date. These adjustments are posted through journal entries to the underlying account with the offset posted to an unrealized gain or loss account. All debit adjustments are offset against the unrealized gain account and all credit adjustments are offset against the unrealized Gain Account and Unrealized Loss Account fields, the net of the adjustments is derived and posted.

For balance sheet accounts, the revaluation journal entries are reversed in the next period. AutoReverse can be used to automate the reversals. For income statement accounts that use the PTD method of revaluation, the revaluation journals aren't reversed since each period's revaluation adjustment is for that period.



In Oracle Fusion General Ledger, the revaluation functionality provides the following advantages:

- Full multicurrency functionality to eliminate currency barriers across a global business.
- · Predefined revaluation rules to ensure consistency in generation of revaluation entries each period.
- Usage of prevailing currency normalization accounting standards including:
 - US Financial Accounting Standards Board (FASB) Financial Accounting Statement No. 52 (FAS 52), Foreign Currency Translation.
 - International Financial Reporting Standards (IFRS) International Accounting Standard No. 21 (IAS 21), The Effects of Changes in Foreign Exchange Rates.
- Support for multiple balancing segments to provide clarity in tracking the profitability and performance for more distinct segments of your enterprise in any currency

Definition

When defining your revaluations, perform the following:

- Include accounts for tracking gains and losses, currency conversion rates, and the number of entered currencies to revalue.
- Define separate revaluation definitions for each class of accounts, using a different rate type for each class.
- Select various conversion types and methodologies for different account ranges, such as:
 - o Current rates and year-to-date (YTD) method for balance sheet accounts.
 - Average rates and period-to-date (PTD) method for income statement accounts.
- Note: Income statement accounts can also be revalued using YTD method.

Hierarchies and flexible account selection criteria, such as usage of parent values from your account hierarchy, streamlines maintenance of revaluation definitions. The parent values can be selected for the primary balancing and the natural account segments using the **Is a last descendant of** operator. Leveraging hierarchy versions extends your revaluation definitions during organizational changes. Adjust account selection criteria monthly to retrieve the accounts that must be revalued for the current accounting period.

Share revaluation definitions across ledgers that have the same chart of accounts to reduce maintenance.

Generation

Generating revaluations include:

- Using defined revaluation criteria and automatically generating entries to shorten your close process.
- Selecting automatic posting as part of the generate revaluation criteria to help you to achieve processing efficiency.
- Scheduling revaluations to run during off peak hours to save your processing resources.
- Using date effective account hierarchies to generate revaluations to keep results in line with your current organization structures.

Always run revaluation to bring monetary balances to current rates before performing currency translation or remeasurement.

Note: When the revaluation process is scheduled to run automatically, the accounting period increments on each subsequent run.



Revaluation Execution Report

The Revalue Balances process automatically generates the Revaluation Execution report when you run revaluation. This report shows the details of your account balance revaluation and the journal batches created after running revaluation. The report includes:

- Currencies and revaluation rates used to revalue your accounts.
- Unrealized gain or loss account in which you recorded net gains and losses.
- Range of accounts revalued.
- Names of your batch and journals that the revaluation process created for each foreign currency.
- Total debits and credits of the created entries.

If the Revaluation process cannot locate rates for one or more currencies, balances are not revalued for those currencies. In this case, the Revaluation process completes with a warning and the execution report lists which currencies are missing rates.

Accounting for Unrealized Gain or Loss on Revaluation: Explained

Revaluation launches a process that revalues the ledger currency equivalent balances for the accounts and currencies you select, using the appropriate current rate for each currency. Resulting unrealized gain or loss amounts are posted to the unrealized gain or loss accounts or to the cumulative translation adjustment account. The revaluation journal is created, balanced, and posted automatically by balancing segment values.

Revaluation journal entries are created to adjust the ledger currency balances for conversion rate fluctuations, in accordance with:

- Statement of Financial Accounting Standards (SFAS) No. 52, Foreign Currency Translation
- International Accounting Standard (IAS) 21, The Effects of Changes in Foreign Exchange Rates

The revaluation journal entries generated and posted in the primary ledger are automatically generated, converted, and posted to each of their reporting currencies. Define the cumulative translation adjustment account in the reporting currency prior to running revaluation.

Income Statement Accounts Revaluation Rule: Explained

Revaluation is the process which adjusts asset or liability accounts that may be materially understated or overstated. The fluctuation in the conversion rate occurs between the time the transaction was entered and the time revaluation takes place. You may want to revalue income statement accounts as well. The Income Statement Accounts Rule indicates whether period-to-date (PTD) or year-to-date (YTD) method is to be used when revaluing income statement accounts.

Click the **Income Statement** radio buttons on the **Create Revaluation** page to revalue income statement accounts using PTD or YTD balances.

If you select to revalue PTD balances for income statement accounts, the process continues to appropriately revalue YTD balances for balance sheet accounts. If the range of accounts consists of both income statement and balance sheet accounts and you select PTD as an option for income statement account revaluation rule, the revaluation:

- Creates separate revaluation journal for the income statement accounts
- Creates weighted average YTD balances using period rates from each corresponding period against the PTD account balance.



Is in compliance with the Statement of Financial Accounting Standards (SFAS) No. 52, Foreign Currency Translation.

When you run revaluation on your income statement accounts, the process produces two separate journal entries; one that revalues your balance sheet accounts and another for your income statement accounts. You do not reverse the PTD revaluation journal for your income statement accounts in the subsequent period. The revaluation only applies to last period's activity.

Note: This functionality only applies when the range of accounts in the revaluation definition consist of income statement and balance sheet accounts. Normally only balance sheets accounts are revalued.

Revaluing Across Multiple Balancing Segments: Worked Example

This example demonstrates how to revalue foreign currency balances across multiple balancing segments. Your company, InFusion America, Inc. has three lines of business. You revalue your foreign currency account balances for two of your divisions, Air Components and Repair Parts. Your Installation Services line of business does not have foreign currency transactions. Your company is your primary balancing segment and your lines of business are represented in your secondary balancing segment.

Note: Enable up to three balancing segments to use the multiple balancing segment feature.

The following are points to consider in running the revaluation process.

- Revaluation posts the resulting gain or loss amounts against the unrealized gain or loss accounts, substituting the balancing segment values appropriately for all balancing segments.
- Gain or loss accounts and revaluation account ranges are not validated against your data access set security when the revaluation definition is created because the ledger context is not known at the time of definition.
- Data access set security is enforced when the Revalue Balances process is executed. Limited write access to the gain or loss accounts due to inadequate access results in an error.
- Segment value security rules are enforced when you enter the account ranges and the unrealized gain and loss accounts. Only segment values you have access to are available in the list of values.
- Account ranges you have read and write access to are revalued. Account combinations that you do not have access
 to are ignored.
- Revaluation expands the parent primary balancing segment to the child values. Data access set security applies to the child values only, not the parent value.
- Posting supports multiple balancing segments for calculating the entry to the Cumulative Translation Adjustment accounts when replicating revaluation journals to reporting currencies.

Defining Revaluations

- Navigator > Setup and Maintenance > Manage Revaluations > Go to Task > Create.
- 2. Enter the values in the following table in the correct fields.

Field	Value
Name	InFusion America Revaluation
Description	Revaluation for all foreign currency balances.
Chart of Accounts	InFusion America Chart of Accounts



Field	Value
Currency	Leave blank
	Note: If left blank, all currencies are revalued and after saving, the field automatically displays: All currencies.
Conversion Rate Type	Daily
Days to Roll Forward	5
Unrealized Gain Account	011-00-96600000-0000-000
Unrealized Loss Account	011-00-96700000-0000-000
Income Statement Account Basis	PTD
Post Automatically	Yes

- 3. In the **Revaluation Accounts** region, click the **Add Row** icon.
- 4. Click the **Change filter conditions** icon to enter the filter used to select the accounts to revalue.
- 5. Click the **Add Field** drop down arrow and select your company, InFusion America Inc. from the list.

Field	Value
Equals	011

6. Click the **Add Field** drop down arrow and select your two Lines of Business: 30 for Air Components and 40 for Repair Parts.

Note: Your Installation Services line of business, 50, is not included because it does not have foreign currency transactions.

Field	Value
Between	30
	40

7. Click the **Add Field** drop down arrow and select Account from the list.

Field	Value
Between	10000000
	29999999



- 8. Click OK to accept your filters.
- 9. Save and Close.

Optionally, select the **Save** and **Generate** buttons to save and run the revaluation immediately.





6 Define Applications Coexistence Configuration

Integration with Oracle E-Business Suite and Oracle PeopleSoft: Overview

Oracle Fusion Applications provides a coexistence strategy that allows you to continue to use your Oracle E-Business Suite or Oracle PeopleSoft General Ledgers and subledgers while using Oracle Fusion Accounting Hub for financial reporting. Coexistence includes the ability to transfer balances from the Oracle E-Business Suite General Ledger and journal entries from Oracle PeopleSoft General Ledger to the Oracle Fusion Accounting Hub.

For more information on completing the post-installation setup for coexistence with Oracle E-Business Suite General Ledger see:

- Configuring Oracle Golden Gate to Integrate the E-Business Suite Ledger with Fusion Accounting Hub on My Oracle Support
- Oracle Fusion Accounting Hub Implementation Guide
- Oracle General Ledger Implementation Guide Release 12.2: See this guide for information on loading and transferring data from Oracle E-Business Suite to the Oracle Fusion Accounting Hub.

For more information on completing the post-installation setup for coexistence with Oracle PeopleSoft General Ledger, see:

- Oracle Fusion Accounting Hub Implementation Guide
- PeopleSoft General Ledger 9.1 Documentation Update: Integrating PeopleSoft General Ledger with Oracle Fusion Accounting Hub
- PeopleSoft General Ledger 9.1 Integration to Oracle Fusion Accounting Hub Implementation Guide
- Note: The Oracle Data Integrator (ODI) component (extract file for manual import) is currently available via My Oracle Support only in note id: 1365971.1.

Register Applications Coexistence Instances

Register applications coexistence instances to indicate in Oracle Fusion General Ledger which Oracle E-Business Suite and Oracle PeopleSoft instances are integrated with the Oracle Fusion Accounting Hub. There is a user interface to this registration. For each E-Business Suite or PeopleSoft instance, provide a unique system identifier. This identifier must also be registered in the corresponding Oracle E-Business Suite or Oracle PeopleSoft instance.

You can specify a unique journal source per instance. For Oracle E-Business Suite, you can limit which instance and balancing segments may post to a particular Oracle Fusion General Ledger.

For Oracle E-Business Suite, determine the Function ID to move data from Oracle Fusion General Ledger to Oracle E-Business Suite General Ledger. You must include the Function ID at the end of the drill down URL that is provided during the registration of the Oracle E-Business Suite instance.



To find the Oracle E-Business Suite Function ID:

- 1. Login as a System Administrator and navigate to Function page
- 2. Query for the function name: GL_FUSION_EBS_DRILL
- 3. Go to the Help menu, click **Diagnostics** > **Examine**
- 4. Select the **FUNCTION ID** field. The value box shows the value of the Function ID.

For Oracle E-Business Suite: The URL format for the non-dynamic portion needs to be in the following format: http://domain>:cdomain>:<pre

In the above URL format, the domain, port, and function_id are for the Oracle E-Business Suite Instance.

For Oracle PeopleSoft: The URL format for the non-dynamic portion needs to be in the following format: http://server/servlet_name/SiteName/PortalName/NodeName/c/PROCESS_JOURNALS.FUS_DRILLBACK_JRNL.GBL

In the above URL format:

- http://server/: Scheme (http or https) and the web server name.
- servlet_name/: Name of the physical servlet that the web server invokes to handle the request.
- SiteName/: Site name specified during Oracle PeopleSoft Pure Internet Architecture setup.
- PortalName/: Name of the portal to use for this request.
- NodeName/: Name of the node that contains the content for this request.

FAQs for Define Applications Coexistence Configuration

How can I sign up for the Oracle Fusion Accounting Hub Reporting Cloud Service?

Proceed to cloud.oracle.com or contact your Oracle account manager.

Define Applications Coexistence Configuration for E-Business Suite

Transfer EBS Balances to Oracle Fusion Accounting Hub Reporting Cloud Service: Explained

The **Transfer EBS Balances to Oracle Fusion Accounting Hub Reporting Cloud Service** is a single Oracle E-Business Suite (EBS) process that performs all the EBS setup necessary to use Oracle Fusion Financial Reporting features.

- The process transfers the following:
 - o All ledgers in a single run.



- All setup components that are needed, such as value sets, values, chart of accounts, calendar, ledger, and hierarchies.
- Actual balances.
- The Transfer E-Business Suite Balances to Oracle Fusion Accounting Hub Cloud Service process:
 - o Create chart of accounts elements such as value sets, segment values, and segment value hierarchies.
 - Create ledgers and calendars.
 - o Open Oracle Fusion accounting periods.
 - Transfer balances to Oracle Fusion by creating and automatically posting journals
 - Monitor the completion status and returns the status to EBS.

Running the EBS Balances to Cloud Reporting Transfer: Examples

To run the Transfer EBS Balances to Oracle Fusion Accounting Hub Reporting Cloud process, follow these steps:

Scenario

- 1. Download the patch from **support.oracle.com**. You can apply this patch on any EBS Release 12 environment and it includes the capabilities needed for the Financial Reporting Cloud.
 - Note: It does not impact your existing applications.
- 2. Login to your Release 12 environment as usual.
- 3. Navigate to Setup > Other > Setup Fusion Accounting Hub Reporting Cloud Service form.
- **4.** Enter the following information:
 - Cloud URL: Use the one assigned to you for your cloud environment.
 - o Cloud user name and password.
 - Select Validate to prevalidate the environment.
- 5. Click Next to proceed to the Autocreate Reports form.
- **6.** Enter the following:
 - Top Revenue Account
 - Top Operating Expense account
 - Top Cost of Sales Account
- 7. Click Submit.
- Note: The first time the Transfer EBS Balances to Oracle Fusion Accounting Hub Reporting Cloud Service process is run, the process:
- Submits an EBS child process that transfers the following:
 - Seed data for all ledgers, including value sets, values, chart of accounts, calendar, and ledger.
 - Actual balances for all ledgers



Note: The initial process can take a couple of hours. You receive an e-mail when it completes.

Calendar Setup From EBS to Oracle Fusion GL: Explained

The calendars and their associated periods are transferred from Oracle E-Business Suite (EBS) to Oracle Fusion General Ledger (GL). As additional periods are added, these periods are also transferred.

FAQs for Integration with Oracle E-Business Suite

When do segment values transfer from EBS to GL?

After a balance had been recorded to the segment value in EBS, the balances transfer process transfers the value to the Oracle Fusion GL.

When are the balances transferred from EBS to Oracle Fusion GL?

Schedule transfers to happen on regular intervals or submit real time processes.

Balance Transfer:

- Transfers the changes to the balances that have occurred since the last balance transfer.
- Automatically processes data for all periods subsequent to the Oracle Fusion GL ledgers first ever opened period, including that period.
- Transfers YTD balances for the first ever opened period of the ledger. For other periods, it transfers changes to PTD balances.
- Maintains segment values as necessary to transfer balances.

What ledger components are not transferred from EBS to Oracle Fusion Accounting Hub Reporting Cloud?

The following components are not transferred:

- Subledger accounting methods
- Legal entities
- Balancing segment value assignments to a ledger or legal entity
- Approval settings
- Intercompany settings
- Journal reversal criteria set
- Transaction calendar



- Average daily balance setting
- Budgets.
- Note: Only primary ledgers are transferred. Any secondary or reporting ledgers related to the primary ledgers are transferred, but appear in Oracle Fusion Accounting Hub Reporting Cloud Service as primary ledgers.

What setup data is not transferred from EBS to Oracle Fusion GL?

The following list provides some of the key setup data that are not transferred. These components are optional and not necessary for the integration.

- Currencies and conversion rates
- Revaluation rules
- Mass allocations
- Cross-validation rules
- Segment value security
- Ledger data access sets
- Note: Since Oracle Fusion Applications has a different security model, Oracle E-Business Suite (EBS) user definitions and data access sets are not transferred.

When are the balances transferred from EBS to Oracle Fusion GL?

Schedule transfers to happen on regular intervals or submit real time processes.

Balance Transfer:

- Transfers the changes to the balances that have occurred since the last balance transfer.
- Automatically processes data for all periods subsequent to the Oracle Fusion GL ledgers first ever opened period, including that period.
- Transfers YTD balances for the first ever opened period of the ledger. For other periods, it transfers changes to PTD balances.
- Maintains segment values as necessary to transfer balances.

What happens if referenced setup data for balances has not been transferred?

The referenced data, such as segment values or open periods, are created by the E-Business Suite (EBS) process in Oracle Fusion General Ledger (GL).



What happens if I drill down from the Inquire on Detail Balances page after an EBS transfer?

The hyperlink drills to the EBS Journal Lines from the **Inquire on Detail Balances** page.

Define Applications Integration with PeopleSoft

Oracle Accounting Hub Integration with Oracle PeopleSoft General Ledger: Overview

The integration between Oracle Accounting Hub and Oracle PeopleSoft General Ledger allows organizations to use Fusion Accounting Hub:

- To transform information from non-Oracle source systems, also known as transaction systems, into journal entries.
- To transfer the journal lines to the PeopleSoft General Ledger for posting.
- Note: The integration uses Oracle Data Integrator and is licensed separately from Fusion Accounting Hub.

A separate integration is provided from PeopleSoft General Ledger to Fusion Accounting Hub for financial reporting.

▼ Tip: For information about the financial reporting integration, see Article ID 1365971.1 on My Oracle Support. You can also access the Oracle PeopleSoft General Ledger Integration to Oracle Fusion Accounting Hub 1.0.1 Implementation Guide and related PeopleSoft documentation updates on My Oracle Support.

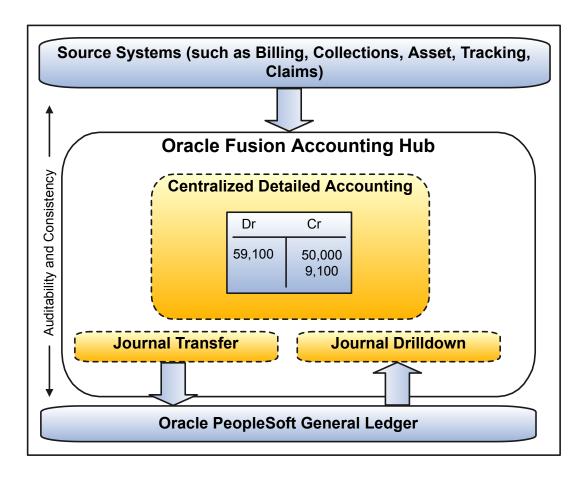
Oracle Accounting Hub Integration Process Flow: Explained

With Oracle Accounting Hub, implementation transactions are fed from non-Oracle source systems. Accounting Hub uses accounting rules, defined during configuration, to determine how to create accounting for these transactions.

Once Accounting Hub accounting is completed, the results are transferred in detail to the Oracle PeopleSoft General Ledger. PeopleSoft General Ledger provides the ability to update the Accounting Hub accounting; primarily to accommodate



combination edit validations and to complete Interunit processing. The PeopleSoft General Ledger posts journal entries to the Accounting Hub. Users can query the results and drill back to Accounting Hub as needed.



Key differences from a standard Accounting Hub implementation and use of Accounting Hub with the PeopleSoft General Ledger are as follows:

- Chart of accounts values (known in PeopleSoft as ChartField values) are loaded from the PeopleSoft General Ledger rather than entered directly into Accounting Hub.
- Accounting Hub journals are not transferred and posted to the General Ledger. Instead, PeopleSoft General Ledger
 is used for financial reporting. Along these lines, translation and secondary ledgers are not used in Accounting Hub.
 Multicurrency and multi-GAAP financial reporting occurs in PeopleSoft General Ledger. The Accounting Hub ledgers
 must, however, still is created since they hold the definition of how the Accounting Hub creates accounting.
- Validation of the chart of accounts values using combination edit rules occurs in PeopleSoft General Ledger.
 Although the users can implement Accounting Hub cross-validation rules, this requires dual maintenance of cross-validation rules and combination edits.
- Accounting Hub does not create intercompany or balancing lines to ensure that the journal balances by balancing segment. These lines are created by the PeopleSoft General Ledger after the Accounting Hub journals are transferred. Interunit processing rules are created in the PeopleSoft General Ledger, and are not available to use in the Accounting Hub. The Accounting Hub journals are balanced by total debits and credits.



Implementation Decisions

ChartFields and Charts of Accounts

The differences between PeopleSoft General Ledger and Accounting Hub chart of accounts values are:

Type	Oracle PeopleSoft General Ledger	Oracle Accounting Hub
Terminology	No single name exists to describe the entirety of all values used to categorize journal lines.	Uses chart of accounts to describe the entirety of all values used to categorize journal lines.
		Example : Corporate chart of accounts can be used to describe a structure of four related components:
		CompanyCost CenterAccountLine of Business
Terminology	Uses ChartField to describe an individual element within the chart of accounts.	Uses segment to describe an individual element within the chart of accounts.
	Examples:	Examples:
	DepartmentChartFieldOperating UnitChartFieldAccount ChartField	Cost Center SegmentAccount Segment
Terminology	Uses SetID plus the ChartField name to describe a group of values for a ChartField.	Uses the name Value Set to describe a group of values for a chart of accounts segment.
	Example : SetID CORP for the Department ChartField holds all the department values for the corporate chart of accounts.	Example : Corporate Cost Centers value set holds all the cost center values for the corporate chart of accounts.
Structure	The structure is set by the PeopleSoft General Ledger. The number of ChartFields is predefined at 16. ChartFields can be enabled or disabled using the ChartField Configuration component.	The structure is defined during configuration. Each chart of accounts can have from two and to thirty segments. The number of segments can vary by ledger.
Effective Dating: Date Range	Multiple effective dates can be used for the same value.	Only one effective date range may be used per chart of accounts segment value in a value set.
	Example: Department 10000 is valid:	Example:
	From January 1, 2010 to December 31, 2010 - manager John Doe.	Cost center 100 is valid:
	AND	From January 1, 2010 - December 31, 2012.



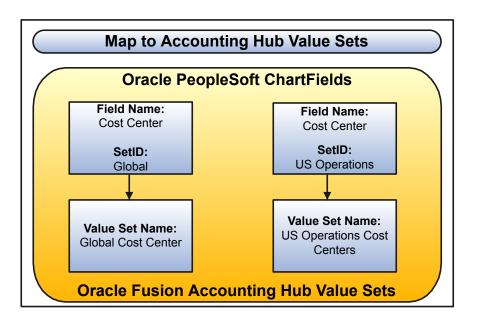
Type	Oracle PeopleSoft General Ledger	Oracle Accounting Hub
	From January 1, 2012 to December 31, 2012 - manager Mary Moon.	
Effective Dating: Meaning	The description for a ChartField value can change over time. The history of the description is maintained.	The description for a chart of accounts segment value cannot change over time.
	Example:	Example:
	Department 10100 has a description of Administration for January 1, 2015 - December 31, 2015. On January 1, 2016, the description is updated to Corporate Overhead. The user can view and report on the different meanings for 10100 based upon the effective date.	Cost center 101 has a description of Administration starting January 1, 2015. On January 1, 2016, the description is updated to Corporate Overhead. The original description of Administration is not maintained in the Accounting Hub.
Assignment	SetID plus ChartField combinations can be assigned for each ChartField in a ledger per business unit, if needed. Otherwise, all business units can use the shared SetID ChartFields.	A single value set can be assigned to each chart of accounts segment for a ledger. The values are not distinguished by business unit.

Key considerations due to differences between PeopleSoft General Ledger and Accounting Hub chart of accounts functionality:

1. Create an Accounting Hub chart of accounts structure that represents the PeopleSoft ChartField structure. As part of the configuration, each segment of the chart of accounts is mapped to a ChartField. You must have a one-to-



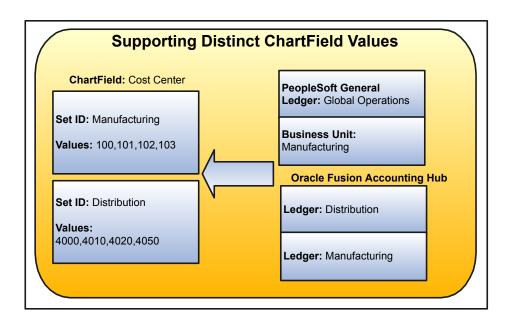
one relationship between the chart of accounts segments and ChartFields to successfully transfer accounting entries from Accounting Hub to the PeopleSoft General Ledger.



- Create a separate value for unique ChartField SetID combination, if shared SetIDs is not acceptable.
- Each Accounting Hub chart of accounts can use only one value set per segment (ChartField).
- PeopleSoft ChartFields are mapped to Accounting Hub value sets.
- 2. Use shared SetIDs to have cross-business unit journals in Accounting Hub. Only one value set can be assigned per segment to the Accounting Hub chart of accounts. This can require a revision to the current ChartField configurations in PeopleSoft General Ledger.



3. Consolidate one or more SetID ChartField definitions so that they can be shared in Accounting Hub



- You can separate Accounting Hub ledgers if the ChartField values are distinct by business, SetID.
- You cannot create cross-ledger journals (transferred to Peoplesoft GL) in Accounting Hub.
- If cross-business unit journals are required in Accounting Hub, the business units must be in the same Accounting Hub ledger.
- 4. Determine description values for the Accounting Hub when updating your ChartField descriptions. Only the most recent effective date range per ChartField value may be transferred to Accounting Hub. The value descriptions appear in the list of values during the Accounting Hub accounting rules configuration such as creation and maintenance of account rules. Configuration of accounting rules determines the chart values used for journal lines.

Creating Oracle Accounting Hub Calendars: Explained

Create Oracle Accounting Hub accounting calendars using Oracle PeopleSoft General Ledger period names. This requires creating more than one Accounting Hub accounting calendar. Accounting Hub ledgers can use only one accounting calendar.

- 1. Transfer the accounting period name from Accounting Hub to PeopleSoft General Ledger with each journal entry.
- 2. The period names must match for the transfer to be successful.

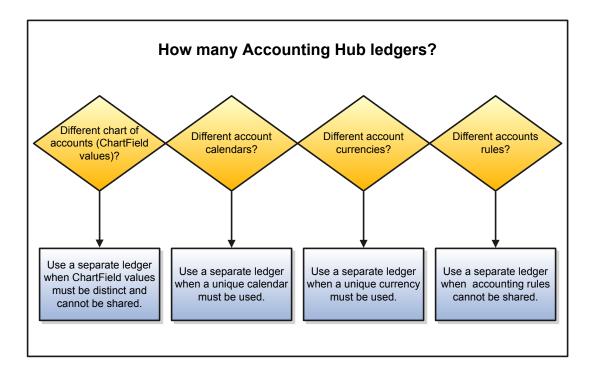
Oracle Accounting Hub Ledgers Process: Explained

Oracle Accounting Hub uses ledgers to create journal entries. The Accounting Hub ledger is used to assign the various accounting rules that transform transaction and reference information into journals. It also determines the period name and chart of accounts values that are used for such journals.



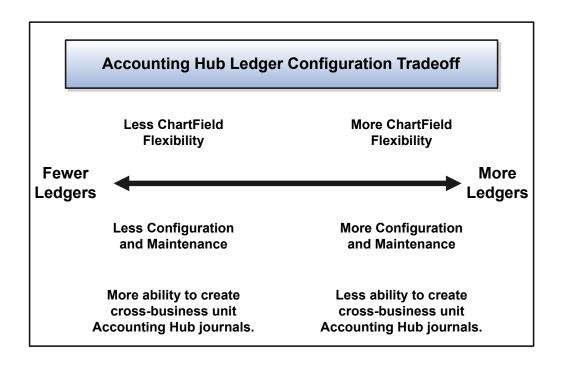
The ledger associated with each Accounting Hub journal and the business unit are used to determine which PeopleSoft ledger is used to book the corresponding PeopleSoft journal. Following are the differences in Oracle PeopleSoft General Ledger and Accounting Hub's use of ledger.

- A one to one relationship may not exist between PeopleSoft and Accounting Hub ledgers
- Several criteria are useful for determining on how many Accounting Hub ledgers are created.



The trade-offs are between the number of Accounting Hub ledgers, functionality, and maintenance. Each Accounting Hub can have a distinct list of values for each chart of accounts segment. You can use this to accommodate the requirement to vary ChartField values by business unit. The trade-offs are that each Accounting Hub ledger requires maintenance; such as opening and closing accounting periods, and the creation of Accounting Hub journals. You can automate opening and closing accounting periods. If your business units are in different Accounting Hub ledgers, then it's impossible to create journals that impact both business units.





ChartField Combination Edits and Cross-Validation Rules

PeopleSoft General Ledger uses Combination Edit Rules to determine the combinations of ChartField values that are acceptable for a journal line. Accounting Hub uses cross-validation rules for this. Significant differences exist in how Combination Edit and Cross-Validation rules are configured and maintained.

Note: Oracle functionality does not provide integration of Combination Edit Rules to Accounting Hub for this integration.

Two options exist for configuration:

- Use PeopleSoft General Ledger combination edit validations after journals are transferred. This option requires less
 configuration and maintenance than simultaneously using Combination Edits and Cross-Validation Rules. However,
 there may be differences between Accounting Hub journal details and PeopleSoft journals. PeopleSoft General
 Ledger maintains histories of journal updates based upon Combination Edit Rules for auditing.
- Use both Combination Edit Rules and Cross-Validation Rules. This option requires additional maintenance, but minimizes the number of updates to Accounting Hub accounting after it's transferred to PeopleSoft General Ledger.

Interunit and Intraunit Processing (Intercompany Accounting)

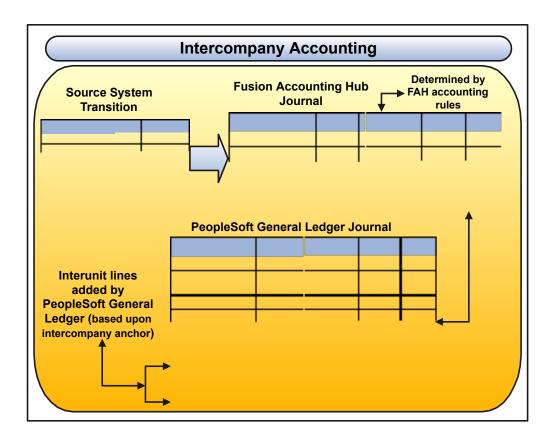
Both applications provide similar functionality. Several conceptual differences exist between Accounting Hub and PeopleSoft General Ledger intercompany configurations. You can enter journals directly into PeopleSoft General Ledger and not in Accounting Hub. Final intercompany processing must be done in PeopleSoft General Ledger.

Interunit configurations in PeopleSoft General Ledger are not transferred to Accounting Hub. Intercompany processing in Accounting Hub is disabled for journals transferred to PeopleSoft General Ledger. Accounting Hub generates the additional journal lines to capture the impact of intercompany accounting.



Note: As part of the accounting rules configuration in Accounting Hub, the anchor line used to determine the Interunit to or from lines, is marked and transferred to PeopleSoft General Ledger. Anchor lines are used by the PeopleSoft General Ledger Interunit processing rules to create the balancing lines.

This process is depicted as follows:



Installation and Integration Configuration: How It's Configured

Installation and configuration are the initial steps to perform for the Oracle Accounting Hub (Accounting Hub) integration with Oracle PeopleSoft General Ledger (PeopleSoft General Ledger).

Settings That Affect Installation Requirement: Critical Step

Ensure that you select Oracle Financials rather than Oracle Accounting Hub, to complete the provisioning wizard.

Configuration Checklist

The following checklist provides the steps required to configure the integration process of Accounting Hub with PeopleSoft General Ledger. The integration process assumes that both the applications are installed, and PeopleSoft General Ledger is configured.

Note: Other steps required for Accounting Hub implementations such as defining accounting rules.



Configuration Steps	Checklist Points	
Configurations completed in both PeopleSoft General Ledger and Accounting Hub	Integrate PeopleSoft ChartFields	
Configurations completed in PeopleSoft General Ledger	 Map external applications to PeopleSoft General Ledger. Map Accounting Hub ledgers to PeopleSoft ledgers. Specify the drill-down URL. Set up an accounting entry definition. Set up the Journal Generator template. Set up ChartField mass change rules (optional). 	
Configurations completed in Accounting Hub	 Complete Accounting Hub security administration. Create Accounting Hub accounting calendars, charts of accounts, and ledgers. Create Accounting Hub subledger applications. Assign Intercompany Anchor Flag accounting attribute. 	
Configurations completed in Oracle Data Integrator	 Create a data server, physical schema, and a logical schema for the log file Create a data server, physical schema, and a logical schema for Accounting Hub Create a data server, physical schema, and a logical schema for PeopleSoft General Ledger. 	

The integration has some specific configuration restrictions that must be implemented or the transfer of journals to be successful:

- 1. The ledger name should not exceed 10 characters.
- 2. The first segment of the chart of accounts (Segment 1) should not exceed 5 characters.
- 3. Conversion rate type names should not exceed 5 characters.
- 4. Journal line descriptions should not exceed 30 characters.
- 5. Currencies should be 3 characters.
- 6. Conversion rates should not exceed 15 numbers, with not more than 8 numbers after the decimal.
- 7. Entered or Accounted amounts used for accounting must not exceed 26 characters, with no more than three numbers after the decimal.

Complete Accounting Hub Security Administration

Submit the journal transfer process to integrate the Accounting Hub security configuration. Use Oracle Authorization Policy Manager (APM) application which is available with the Accounting Hub installation.

- Note: The user must have administrator credentials for APM.
- 1. Sign in to the Authorization Policy Manager.
 - Select fscm Application Nameselection box and select Search Users.
- 2. Assign Subledger Accounting Integration Duty Role.
 - Search for the user who is the assignee of the role, and select Open User.
 - Select the Application Role Assignments tab and select Map icon
 - Search using the following:



Search Field	Value
Application Name	fscm
Display Name	Subledger Accounting Integration Duty Role

- 3. Select Subledger Accounting Integration Duty role from the search results.
- 4. Select Map Roles to complete the mapping.
- **5.** Expand **fscm** to verify the role mapping is complete.

Synchronize Financials Application Identifier: Procedure

The procedure details the configurations to synchronize the financials application identifier (APPID).

Prerequisites

Following are the prerequisites to synchronize the APPID:

- 1. Obtain the Oracle database details.
- 2. Obtain the password for FUSION_ODI schema.
- 3. Get the USER_GUID for APPID user FUSION_APPS_FIN_ODI_ESS_APPI, using the following query:

SELECT USER GUID FROM FUSION.PER USERS WHERE USERNAME = 'FUSION APPS FIN ODI ESS APPID';

Procedural Steps

Following are the procedure steps to synchronize the APPID:

- 1. Sign in to the host machine where Oracle Accounting Hub is installed.
- 2. Ensure Java run time environment is installed on your system. You can verify this by running java-version command from the host terminal.
- **3.** Run the following commands in sequence:
- Create APPID in Oracle Data Integrator Master Repository

\$APPTOP/fusionapps/odi/modules/oracle.jdbc_11.1.1/ojdbc6dms.jar:\$APPTOP/fusionapps/odi/setup/manual/oracledi-sdk/oracle.odi-sdk-jse_11.1.1.jar oracle.odi.util.odiConfigCreateAppId FUSION_ODI <
Password for FUSION_ODI schema user> "jdbc:oracle:thin:@<fusion database host>:<fusion database port>/
<fusion database SID>"FUSION APPS FIN ODI ESS APPID

Synchronize GUID for APPID with External LDAP

\$APPTOP/fusionapps/odi/modules/oracle.jdbc_11.1.1/ojdbc6dms.jar:\$APPTOP/fusionapps/odi/setup/manual/oracledi-sdk/oracle.odi-sdk-jse_11.1.1.jar oracle.odi.util.odiConfigCreateAppId FUSION_ODI <
Password for FUSION_ODI schema user> "jdbc:oracle:thin:@<fusion database host>:<fusion database port>/<fusion database SID>"FUSION_APPS_FIN_ODI_ESS_APPID<USER_GUID for APPID FUSION APPS FIN ODI ESS APPID



Creating Calendars, Charts of Accounts, and Ledgers: Explained

You can complete the Oracle Accounting Hub configurations using the Setup and Maintenance work area, accessible from the Navigator. Create an implementation project and assign the Accounting Hub task list. The Accounting Hub task list includes all the tasks required to complete the required configurations.

Important: Segment 1 of the Accounting Hub chart of accounts must be configured to support storing business unit values for transfer to Oracle PeopleSoft General Ledger.

Accounting Calendars

No integration feature exists to create Accounting Hub calendars based upon PeopleSoft General Ledger configurations. Use the standard accounting calendar task as part of Accounting Hub implementation project to create accounting calendars as needed. Match the period names in the Accounting Hub calendar to those in the PeopleSoft ledger where the Accounting Hub journals are posted.

Charts of Accounts

Use the standard Accounting Hub functionality to create charts of accounts.

- Each Accounting Hub ledger requires a chart of accounts. The chart of accounts can be used for several ledgers if the values for each segment are the same.
- The structure for all chart of accounts should replicate the PeopleSoft General Ledger ChartField definition. Each
 enabled ChartField should be captured as a separate chart of accounts segment.
- To create a chart of accounts, first create a value set for each segment.
 - Value sets can be shared across multiple charts of accounts, but there is no ability to vary the values.
 - In PeopleSoft values may be segregated by business unit or SetID. If distinct values are required, and cannot be shared, then use separate value sets.
- Important: Segment 1 of the Accounting Hub chart of accounts must be used to capture the business unit information transferred to PeopleSoft General Ledger. The value set names are used on the Value Set Mapping page.

Ledgers

Create a separate Accounting Hub ledger for each unique combination of chart of accounts and accounting calendar. The currency must match the corresponding PeopleSoft ledger. Only primary ledgers should be created. Other ledger types such as secondary ledgers are not eligible for transfer to PeopleSoft General Ledger.

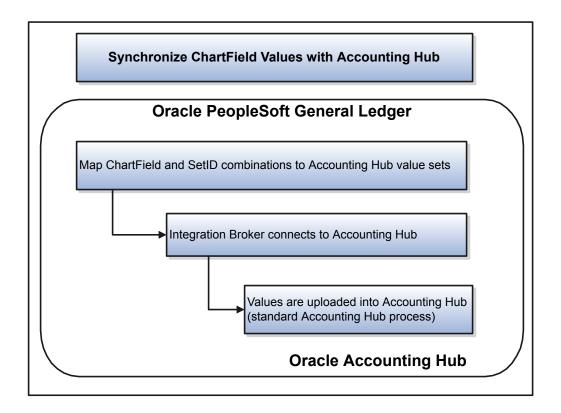
A data access set is created automatically for each Oracle ledger. Use this to submit the Accounting Hub process to import the PeopleSoft ChartField values.

Integrate PeopleSoft ChartField Values: Explained

Oracle PeopleSoft General Ledge ChartField values are integrated with the Oracle Accounting Hub. The integration helps the users to continue using the PeopleSoft General Ledger to enter and maintain chart values.



You can use this integration process for both Accounting Hub configuration and to synchronize the updates to ChartField values such as updates. The updates include changes to fields such as an effective date or ChartField value description.



Note: You must first define the Accounting Hub ledgers to integrate the PeopleSoft General Ledger ChartField values.

Following are the two steps involved in the integration process:

1. Map ChartField and SetIDs Combinations to Accounting Hub Value Sets

Complete the mapping of ChartField and SetID combinations to Accounting Hub value sets.

Following are the Value Set Mapping (ChartField and SetID Combination Mapping) field descriptions.

Value Set Mapping

Field	Value
Field Name (filter)	Select to filter the ChartFields for greater efficiency in updating and viewing.

Value Set Mapping Grid



Field	Value
Field Name	Select a PeopleSoft ChartField that you want to map to a Fusion Value Set Code.
SetID	Select the PeopleSoft SetID , which, in combination with the selected PeopleSoft ChartField, maps to the corresponding Fusion Value Set Code .
Fusion Value Set Code	Enter the unique Fusion Value Set Code that maps to the corresponding PeopleSoft SetID and ChartField combination.
	Note: You must enter the Fusion Value Set Code exactly as it exists within Accounting Hub. The code is case-sensitive.

2. Export PeopleSoft ChartFields

Export the ChartField values for each SetID and ChartField combination that is to be transferred to Accounting Hub. Complete this process by using the **Export ChartFields to Accounting Hub** page in the PeopleSoft General Ledger.

Following are the field descriptions from the **Export ChartFields to Accounting Hub** page.

Field	Value
Process Frequency	Select one of the following values when processing the Export ChartFields to Oracle Accounting Hub batch process request:
	Don't Run: Ignores the request when the batch process runs.
	 Always: Executes the request every time the batch process runs.
	 Once: Executes the request the next time the batch process runs. After the batch process runs, the process frequency automatically reverts to Don't Run.
Request Date Option	Select one of the following:
	 As of Date: Uses the date that you specify to select the effective-dated ChartField values to export.
	 System Date: Uses the current system date to select the effective-dated ChartFields to export.
As of Date	If you select the Request Date option, specify the As of Date to be used in selecting the effective-dated ChartField values to export.
Field Name	Select the ChartField that you want to export.
	Note: A separate row should be completed for each unique combination of ChartField and SetID to be exported to Accounting Hub
External System ID	Select the user-defined External System ID on the External System Setup page. The page includes:
	PeopleSoft Journal Template name.
	 Information related to processing options.
	 Drill back URL for access to view the entries in Accounting Hub.



Field	Value	
SetID	Select the name of the PeopleSoft SetID you want to export.	
Look Back Days	(Optional) Select to export the values as on a specific date.	
File Name	Specify the name to use for the file. Use the suffix .csv for comma separated values.	
Include Related Languages	If you want the descriptions of the ChartField values in alternate languages to be exported, select Yes . These languages must also be installed for Accounting Hub.	
	Note: Only installed languages where there are descriptions in the alternate languages are exported.	
Export All Values	Select to export all ChartField values for the ChartField and SetID combination.	
	Note: After you export the values, the Accounting Hub process is used to complete the transfer.	

In earlier releases:



- Release 8: The terms used for the ChartField integration are Load Interface File for Import in the related Accounting Hub processes. The last step is to submit the process, Import Segment Values and Hierarchies. These processes can be scheduled to run at regular intervals.
- 2. Release 9: Accounting Hub process Load Multiple Interface Files for Import process is submitted. The process automatically runs a second request called: General Ledger Segment Values and Hierarchies Import Process Launcher. No user action is used to submit this process.

Accounting Rule Configurations: Procedure

The following configurations are completed as part of the definition of each subledger application in Oracle Accounting Hub.

- 1. Transfer to PeopleSoft General Ledger profile option: Enable the profile option to display the Transfer to PeopleSoft option for subledger applications.
- 2. **Subledger Applications for Transfer to PeopleSoft**: Enable the PeopleSoft transfer options for a subledger application to ensures that:
 - Validation runs.
 - The PeopleSoft Anchor Flag is assigned during the accounting rules configuration.
 - Journals are to be transferred to PeopleSoft General Ledger instead of being posted to the Oracle General Ledger.
 - Note: Navigate to: Accounting Hub Setup and Maintenance work area > Manage Subledger Application task.



3. Intercompany Driver Line Indicator accounting attribute:

Assign this Accounting Hub accounting attribute created for this integration. A source must be assigned to this accounting attribute so that PeopleSoft General Ledger can identify the journal line which is the anchor. Accounting Hub intercompany balancing is disabled for subledger journals created for applications marked as transferred to PeopleSoft. The intercompany balancing is completed in PeopleSoft General Ledger, based upon the PeopleSoft intercompany configurations.

Note: Navigate to the Accounting Hub Setup and Maintenance work area > Manage Accounting Attributes task.

The Intercompany Driver Line Indicator can have two possible values \mathbf{Y} for Yes, or \mathbf{N} for No. The journal line marked as Yes is used to as the anchor for PeopleSoft intercompany balancing.

Complete PeopleSoft Security Administration

Configuration of the integration requires use of both new and old PeopleSoft General Ledger pages. The following lists the PeopleSoft General Ledger pages required to complete the configurations and process Accounting Hub accounting. Use the **Process Journals** menu.

Page Title	Component	Page
Export ChartFields to Accounting Hub	CF_EXP_REQ	CF_EXP_REQ
ChartField Export Tables	CF_EXP_TBLS	
Fusion Value Set Mapping	CF_VALUESET_MAP	
External Accounting Entries	FAI_ACCTG_ENTRIES	
Account Type Mapping	FAI_ACCTYPE_MAP	
External System Setup	FAI_EXTSYS_SETUP	
External System Setup - Field Mapping	FAI_EXTSYS_SETUP	
Drill to Source	FAI_GL_DRILL	
Fusion Ledger Mapping	FAI_LEDGER_MAP	
ChartField Mass Change Rule Definition	FAI_RULES_DEFN	
ChartField Mass Change Rule Group	FAI_RULES_GROUP	
External Accounting Integration Workbench	FAI_WORKBENCH	
3rd Party Integration Edit	RUN_FAIJEDIT	

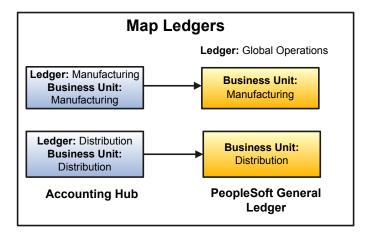


Page Title	Component	Page
3rd Party Integration Post	RUN_FAIJPOST	
External Accounting Integration Error Report	RUN_GLFAI001	
Combo Edit Error Report	RUN_GLFAI002	

Map Ledgers

You may map multiple Accounting Ledgers to the same PeopleSoft General Ledger if the Accounting Hub has different:

- Chart of accounts configurations.
- Accounting rules.
- Note: This mapping is required when you use different chart of accounts values, by business unit, which outweighs the importance of creating Accounting Hub journals.



- Multiple Accounting Hub ledgers can map to the same PeopleSoft ledger.
- Typically map to a primary PeopleSoft ledger.

Use the Fusion Ledger Mapping - Ledger Mapping page to map Accounting Hub ledger and business unit combinations to PeopleSoft ledgers. A mapping must be completed for each Accounting Hub ledger to successfully transfer the related accounting to the PeopleSoft General Ledger.

Field	Value
Business Unit	Select the PeopleSoft business unit of the ledger.
Ledger	Select the PeopleSoft ledger for the corresponding business unit.
Fusion Ledger Name	Enter the ledger name exactly as it exists in Accounting Hub.



Specify Drill-Down URL

The **URLID** provides a link from the PeopleSoft General Ledger to the Accounting Hub. The value used varies based upon your implementation of Accounting Hub. The URL is used to drill from the PeopleSoft General Ledger to the Accounting Hub. Set this option in the PeopleSoft General Ledger.

Hypertext Transfer Protocol	Oracle Ledger Uniform Resource Locator (URL)	Location
Set by your administrator.	The URL to the Journals work area.	The locator ends with / ledger/ faces/ JournalEntryPage
		Note: The question mark symbol is required. The PeopleSoft General Ledger appends the identifier information to the URL during drill-down to obtain the correct Accounting Hub journal entry.

- 1. **Market Rate Type**: If the conversion rate type of **User** is used by the Accounting Hub, then create the same rate type in the PeopleSoft.
- 2. External System Setup page: Defines the External System entity for the PeopleSoft GL external accounting integration. The External System setup for Accounting Hub is preconfigured and delivered. However, you must update the UCM Server Information region according to the UCM (Universal Content Management) environment for the ChartField integration. Review all other fields and update according to specific integration needs.

Field	Value
Description	External System Description.
Header Record	Select the External System Header Interface Record.
Line Record	Select the External System Line Interface Record.
Journal Template	Select the Journal Generator Template with the appropriate SetID.
Last Sequence Number	An automatic sequence number that increases according to the transactions processed in the integration.

Process Options

Field	Value
Edit ChartField Combinations	Select to enable ChartField editing rules to be validated against Accounting Hub entries.
Enable ChartField Mass Change	Select to modify ChartField values after Accounting Hub journals are transferred to PeopleSoft General Ledger. Based upon rules and criteria, ChartField values can be changed.



Field	Value
Create Interunit and Intraunit Entries	Determines whether PeopleSoft General Ledger should create Interunit and Intraunit entries for the Accounting Hub journals.

UCM Server Information

Field	Value
UCM Server URL	Enter the Universal Content Management URL.
User ID	Enter the UCM Server User ID
Password	Enter the UCM Server encrypted Password. Use the Password Encryption tool (Password Encryption group box below) to encrypt the password.

Drill back Information

Field	Value
URL Identifier	Select the Accounting Hub URL to be used for drill back. You can use the predefined value.
Drill back URL Parameters	Accounting Hub uses a list of parameters in the URL for the drill back link to access the selected transaction. Use this grid to insert the required parameters and the corresponding values from the available fields.

Review External System Setup Field Mapping

The fields from the External Accounting Integration record must be mapped against the available fields in the Accounting Hub line record. The External System Setup - Field Mapping page is delivered with preconfigured values. However, as part of configuration, review and update the mapping according to your installation configuration.

Following are the preconfigured field name values and mapping requirements:

Target Field Name in PeopleSoft	Preconfigured Value (Fusion Accounting Hub)	Required
ACCOUNT	SEGMENT2	Yes
ACCOUNT_DT	ACCOUNTING_DT	Yes
AFFILIATE		
AFFILIATE_ INTRA1		
AFFILIATE_ INTRA2		
ALTACCT		



Target Field Name in PeopleSoft	Preconfigured Value (Fusion Accounting Hub)	Required
BUDGET_REF		
BUSINESS_UNIT	ANCHOR_BU	Yes
BUSINESS_ UNIT_CHRG	BUSINESS_ UNIT_CHRG	Yes
CHARTFIELD1		
CHARTFIELD2		
CHARTFIELD3		
CLASS_FLD		
CURRENCY_CD	BASE_CURRENCY	Yes
DEPTID	SEGMENT4	Yes
FOREIGN_AMOUNT	FOREIGN_AMOUNT	Yes
FOREING_CURRENCY	FOREIGN_ CURRENCY	Yes
FUND_CODE	SEGMENT7	
IU_ANCHOR_FLG	IU_ANCHOR_FLG	Yes
IU_TRAN_GRP_NBR		
JRNL_LN_REF		
LEDGER	LEDGER	Yes
LINE_DESCR	DESCR	Yes
MONETARY_ AMOUNT	MONETARY_ AMOUNT	Yes
OPERATING_UNIT		
PRODUCT		
PROGRAM_CODE		
PROJECT_ID		



Target Field Name in PeopleSoft	Preconfigured Value (Fusion Accounting Hub)	Required
RATE_DIV	RATE_DIV	Yes
RATE_MULT	RATE_MULT	Yes
RT_TYPE	RT_TYPE	Yes

Accounting Entry Definition: Explained

Oracle PeopleSoft delivers an Accounting Entry Definition to select data. The Journal Generator stores this data from third-party applications. The Journal Generator uses the Accounting Entry Definition, FAIJRNL (financial accounting integration journal), to select the accounting entries from the Oracle Accounting Hub and to create the PeopleSoft General Ledger journals.

The Accounting Entry Definition, FAIJRNL, is defined under the **SHARE SetID**, following existing PeopleSoft Subsystem Accounting Entry Definition TableSet control setup. This predefined accounting entry definition is based on the external source system mapping. You can use the Journal Generator request page to create PeopleSoft General Ledger journals.

Note: Accounting Entry Definition does not require updates unless the provided external system mapping is changed.

The following table describes the accounting entry definition based on the external source system mapping:

Name	Value
SetID	Control data that is common to all business units is defined within the SHARE SetID , which is used by all business units.
Accounting Definition	Predefined Accounting Entry Definition, FAIJRNL, (financial accounting integration journal).
System Source	PeopleSoft delivers the Journal Generator, External Accounting Integration value to be used for this integration. The system source identifies the source of an accounting entry from which a transaction arises. Journal Generator uses system source to:
	Derive the prompt table for the application business units when it defines the request.Format the system source field on the journal header.
Page Name and Cross Product Drill- Down	Enter the Accounting Entry Page Name FAL_GL_DRILL and enable the Cross Product Drill-Down check box. The Journal Generator process reads this definition to:
	 Create journal entries from the Accounting Hub accounting entries. Enable drill back from the Journal Entry Inquiry page.



Oracle PeopleSoft Integration: How It's Configured

The following procedure illustrates the configurations from the Oracle PeopleSoft to the Oracle Accounting Hub. These configurations are typically completed by a database administrator rather than a functional implementor.

Update PeopleSoft Database TNS File

The TNS names file for the database used by PeopleSoft General Ledger must be edited to include the following entry:

```
ORACLE_FAH_DB= (DESCRIPTION=
(LOAD_BALANCE=YES)
(FAILOVER=YES)
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=tcp) (HOST=Fusion Accounting Hub Host) (PORT=Fusion Accounting Hub Port))
)
(CONNECT_DATA=
(SID=Fusion Accounting Hub SID)
)
```

Note: The TNS names file is used by the Oracle Data Integrator to link the PeopleSoft General Ledger and Accounting Hub databases.

Configure Oracle Data Integrator

The PeopleSoft and Accounting Hub integration uses Oracle Data Integrator. The Data Integrator is:

- A data integration platform used to transfer Accounting Hub journals to the PeopleSoft General Ledger.
- Included as part of the Accounting Hub installation.

The general components listed below must be completed as part of the configuration.

- 1. Configure the log file location for Oracle Data Integrator. The log file is used to store exception messages documenting issues with the transfer of journals from Accounting Hub to PeopleSoft General Ledger.
- 2. Configure the Oracle Data Integrator target, which is PeopleSoft General Ledger. The target is used for the inbound Accounting Hub journals to PeopleSoft General Ledger.

Each of these configurations requires specification of the data server as well as the physical and logical schemas.

- The data server describes connections to the physical databases, known as the data source. It contains details for the host, port, user, and password parameters for the database. The log file uses a local server path which is specified during the data server configuration.
- The physical schema describes the objects used to hold the Accounting Hub journals. Use the physical schema
 to specify which database schema is used. For example, for Accounting Hub it's the FUSION schema and for
 PeopleSoft it's the EMDBO schema.
- The logical schema holds the relationship between the physical schema and the context. It can be used by multiple users, each with a different context.



Configure PeopleSoft Integration Broker

The PeopleSoft integration broker is used to integrate the ChartField values with Accounting Hub. The following configuration steps are required to ensure this process functions.

- 1. **Ping the Gateway**: Query the gateway. Be sure that the RIDCTARGET connector is listed and the gateway is up and running. The RIDCTARGET connector is delivered as a PeopleTools hot patch for customers using PeopleTools 8.53.
- 2. Review the Integration Setup Services:
 - Review the Service Operation FAI_UCM_UPLOAD.
 - Select the link entitled, Service Operation Security.
 - Configure the permission policies.

Review the Node Definition

Review the node definition as given below:

- Query the nodes and review the node UCM_NODE. If it's not active select the row for UCM_NODE_OUTPUT and then
 the Activate Selected Routings button.
- Click the Active Node box and select the Routings tab.
- Confirm that the routing definition the UCM_NODE_OUTPUT routing has Active status.

Configure the External System Setup

Use the following steps to configure the external system setup:

- Use the External System Setup page to enter information for the Universal Content Manager (UCM).
- Complete the UCM Server URL, and password.
- The password must be encrypted. You can use the Password Encryption tool to encrypt your password.

Provide Database Grants

Grant permission to the fusion_odi_stage schema to access fusion.gl_daily_conversion_types, fusion.xla_subledgers, fusion.gl_je_sources.

Here is a sample command.

```
GRANT SELECT ON FUSION.GL_DAILY_CONVERSION_TYPES TO FUSION_ODI_STAGE;
GRANT SELECT ON FUSION.XLA_SUBLEDGERS TO FUSION_ODI_STAGE;
GRANT SELECT ON FUSION.GL JE SOURCES TO FUSION ODI STAGE;
```

Configure Oracle Data Integrator: Procedure

As part of this integration, Oracle Development has predefined the names of the data server, physical schema, and logical schemas. Distinct names are used for the log file, source, and target. However, each name is repeated for all three components (the data server, physical schema and logical schema).

- FILE OUTPUT FIN is used for the log file.
- ORACLE FUSION is used for Oracle Fusion Accounting Hub.
- ORACLE_PSFTis used for PeopleSoft General Ledger.



The following table summarizes the configuration requirements for Oracle Data Integrator for this integration.

Connection Type	Data Server	Physical Schema	Logical Schema
Log File	Must be created.	Must be created.	Must be edited to provide.
PeopleSoft General Ledger	Must be created.	Must be created.	Must be edited to provide context.

Detailed integration process is described in the following procedure:

1. Sign in to Oracle Data Integrator

The URL to access Oracle Data Integrator is specified during the Oracle Fusion Accounting Hub installation. The user name and password are also established during the Oracle Fusion Accounting Hub installation.

- Note: Your help desk can provide the URL and sign in credentials to access Oracle Data Integrator.
- The repository contains Oracle Data Integrator artifacts. Use the Work Repository option to complete the configurations.
- Use the Browse tab, and navigate to Topology > Data Servers.

2. Create a Data Server for the Log File

- Select the Data Servers navigation node from the Browse window by selecting Topology > Data Servers.
- Select the Create icon to create a new data server.
- Enter the following fields on the Add Data Server page:

Data Server Name	This value is used by the integration process for logging exception messages. It must be entered exactly as stated here. It's case-sensitive.
Technology	File
	This indicates that the messages are stored in a file and for other purposes as needed by Oracle for the integration.
Batch Update Size	Default size 30.
Array Fetch Size	Default size 30.
JDBC Driver	com.sunopsis.jdbc.driver.file.FileDriver
JDBC URL	jdbc:snps:dbfile

Select Save to finish creating the data server.



3. Create a Physical Schema for the Log File

The physical schema holds the objects where log messages are stored. Select Physical Schema from the browse pane under the Schemas node.

Select the create icon and select Data Server > FILE_OUTPUT_FIN from the Create Physical Schema page to enter the below values.

Note: The physical schema name automatically refreshes to match the data server name FILE_OUTPUT_FIN.

Field	Value
Physical Schema Name	FILE_ OUTPUT_FIN
	(Automatically populates once the data server is selected.)
Data Server	FILE_ OUTPUT_FIN
Schema Name	/tmp/ODI_IN
Work Schema Name	/tmp/ODI_IN

Note: All the remaining fields have default values and it isn't required to update. Select **Save** to finish creating the physical schema for the log file.

4. Edit the Logical Schema for the Log File

The logical schema must be updated to specify the context. Navigate: Logical Schemas > FILE_OUTPUT_FIN, and select the edit icon.

Context: Development	FILE_OUTPUT_FIN/tmp/ODI_IN	
Context: Global	FILE_OUTPUT_FIN/tmp/ODI_IN	

5. Create a New Data Server for PeopleSoft General Ledger

Select the Data Servers navigation node from the Browse window by selecting Topology > Data Servers. Select the create icon to create a new data server.

Enter the values in the Add Data Server page:

Field	Value
Data Server Name	This value is used by the Accounting Hub transfer process. It must be entered exactly as stated here. It's case-sensitive.



This indicates that PeopleSoft General Ledger is installed on an Oracle database. Currently, this integration requires that the PeopleSoft General Ledger is installed on an Oracle database. Default size 30. Default size 30. oracle.jdbc.driver.OracleDriver jdbc:oracle:thin: <host>:<port>:<sid></sid></port> is the name of the host used by PeopleSoft General Ledger for your installation.</host>
Default size 30. oracle.jdbc.driver.OracleDriver jdbc:oracle:thin: <host>:<port>:<sid></sid></port></host>
oracle.jdbc.driver.OracleDriver jdbc:oracle:thin: <host>:<port>:<sid></sid></port></host>
jdbc:oracle:thin: <host>:<port>:<sid></sid></port></host>
<host> is the name of the host used by PeopleSoft General Ledger for your installation.</host>
and the field of the free cook and field the field and the
<port> is the name of the port used by PeopleSoft General Ledger for your installation.</port>
<sid> is the site identifier used by PeopleSoft General Ledger for your installation.</sid>
Note: These values are not the same as the data server for Accounting Hub. This is the connection to PeopleSoft General Ledger for the Oracle Data Integrator transfer.
This name is specified as part of PeopleSoft General Ledger installation. It must be a user who has read write access to the tables:
Note: The user must have permission to create, update, and drop tables created at runtime.
Enter your JDBC password. This was specified during the PeopleSoft General Ledger installation, but may have been updated by your database administrator.
ALTER SESSION SET NLS_LENGTH_SEMANTICS=CHAR

6. Create a Physical Schema for PeopleSoft General Ledger

This physical schema holds the objects for Accounting Hub journals after their transfer to PeopleSoft General Ledger. It points to the data server ORACLE_PSFT.

- Select Physical Schema from the browse pane under the Schemas node.
- Select the Create icon to create physical schema.
- Select Data Server from the Create Physical Schema page. Complete the following options:

Field	Value
Physical Schema Name	ORACLE_PSFT



Field	Value
	(Automatically populates once the data server is selected.)
Data Server	ORACLE_PSFT
Schema Name	EMDBO
Work Schema Name	EMDBO

- Note: All the remaining fields have default values and it isn't required to update.
- Click **Save** to finish creating the physical schema for PeopleSoft General Ledger.

7. Edit the Logical Schema for PeopleSoft General Ledger

Edit the logical schema for the PeopleSoft General Ledger. Navigate to Logical Schemas > ORACLE_PSFT, and select the edit icon.

Field	Value
Context: Development	ORACLE_PSFT.EMDBO
Context: Global	ORACLE_PSFT.EMDBO

Accounting Processes: Explained

The following steps are completed at regular intervals, based upon operational considerations such as how often accounting is completed and the accounting close.

Accounting Process	Operational levels
Accounting Hub Processing	 Create Accounting Hub detailed accounting. Transfer Accounting Hub to PeopleSoft General Ledger
PeopleSoft General Ledger Processing	 Submit the PeopleSoft 3rd Party Integration Edit Process Request. Submit the PeopleSoft 3rd Party Integration Post Process Request. Inquire on PeopleSoft journals from the External Accounting Integration Workbench Complete Optional Steps: Run Integration Error Report. Run Combo Edit Error Report if combo edits are enabled. Submit the PeopleSoft Generate Journals Request



Oracle Accounting Hub Accounting Processes: Explained

The Oracle Accounting Hub accounting process is involved in the following two steps:

- 1. Create Accounting Hub Accounting
- 2. Transfer Accounting Entries to PeopleSoft General Ledger

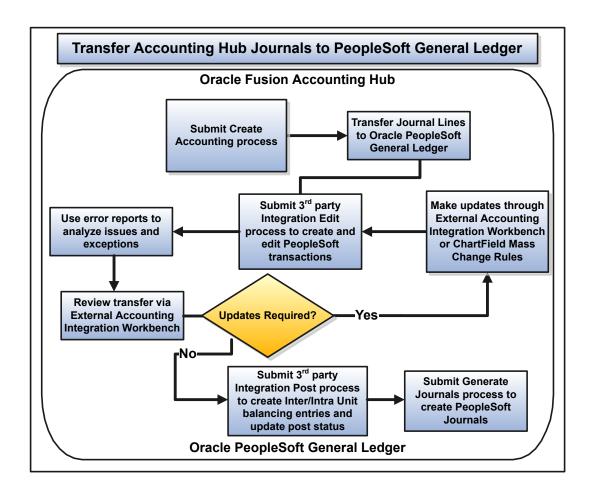
Create Accounting Hub Accounting

The standard Accounting Hub process should be used to create accounting. Once the journals are created (as Final), they are eligible for transfer to PeopleSoft General Ledger. Draft journals are not eligible for transfer, and also, only journals with a balance type of Actual are eligible for transfer.

Transfer Accounting Entries to PeopleSoft General Ledger

A separate Accounting Hub process is used to transfer journals to PeopleSoft General Ledger. The process is submitted automatically when you create Accounting Hub accounting for a subledger application that has accounting that is to be transferred to PeopleSoft General Ledger.

The exception management process, such as Review transfer via External Accounting Integration is optional in the case when there are no errors.





Submit PeopleSoft Integration Edit Process: Explained

Use the 3rd Party Integration Edit run control page to launch the 3rd Party Integration Edit process.

Once the process is complete, you can check the Application Engine message log within the Process Monitor, which includes the number of accounting entries processed and number of entries with errors.

Field	Descriptions
Process Frequency	 Select one of the following values when processing the request: Process Once - executes the request the next time the batch process runs. After the batch process runs, the process frequency automatically reverts to "Don't Run". Always Process - executes the request every time the batch process runs. Don't Run - ignores the request when the batch process runs.
External System ID	Select the user-defined External System ID (External System Setup page. This includes the PeopleSoft Journal Template name used by the Journal Generator and information related to processing options and Drillback URL for access to view the entries in Oracle Accounting Hub. System ID corresponds to any 3rd party System that integrates with PeopleSoft GL. This setup is required to map fields from a 3rd party system to PeopleSoft GL (for example: business unit, ledger, currency, ChartField values, and so on) and the field that identifies the Journal Header Transaction ID.
Post Transactions	Select this check box to post accounting entries by calling the 3rd Party Integration Post process. Posted transactions can be selected by the Journal Generator process to create PeopleSoft General Ledger journals.
From Accounting Date To Accounting Date	Select the range of dates to be used for selecting accounting data from 3rd Party records for processing. Each 3rd party system accounting entry has an accounting date that is accessed for filtering data to be processed.
Business Unit	Select the business unit or units to be processed for this request. At least one business unit is required
ChartField Mass Change Group	Select the ChartField Mass Change Group for applying ChartField Mass Change Rules based on rule criteria.

You can schedule a request periodically to pick up the entries that are generated by any 3rd party system (Oracle Accounting Hub), to import, validate, process, and prepare the accounting entries to be run by the Journal Generator process for PeopleSoft journal entry creation.

This batch process performs the following steps as depicted below:

Validate Accounting Entries

Third Party Integration runs required validation to prevent unnecessary processing, such as:

- Balancing Check: Ensures that transaction header journals are balanced (debits equal credits for business unit, base currency code, book code, adjustment type, and transaction header ID).
- Accounting Date validation: Ensures that Header and Line transactions have the same accounting date.



- Anchor Flag validation: Ensures that the Anchor Flag value is Yes for only one accounting line per transaction header.
- Note: Transactions with Integration Validation errors are excluded from the Edit process and listed on External Accounting Integration Error report.

Insert Accounting Entries into the External Accounting Entry tables

After all validations are completed, the valid accounting entries are inserted into the External Accounting Entry tables (Header and Line) that are used by Journal Generator to create PeopleSoft General Ledger journals. The External System Mapping is used to map the fields at the time the accounting entries are inserted into the External Accounting entry records.

Run ChartField Combination Editing

The 3rd Party Integration Edit process calls the PeopleSoft General Ledger Combination Edit process (FSCOMBO_LIBR) to validate ChartField combinations for all generated accounting entries. The ChartField Combination Editing Template, FAIJEDIT, is used to call the Combo Edit process. PeopleSoft delivers this template as system data and it contains the structure definition for External Accounting transaction entries. Once completed, this process generates a report indicating combination editing errors during processing for user review and action.

Apply ChartField Mass Change Rules

This integration provides the ability to create ChartField Mass Change Rules based on field values and status to change ChartField values in mass. Based on the Mass Change Group that is defined on the 3rd Party Integration Edit run control page, the process analyzes data and applies ChartField Mass Change Rules as according to the processing sequence defined in the ChartField Mass Change Groups.

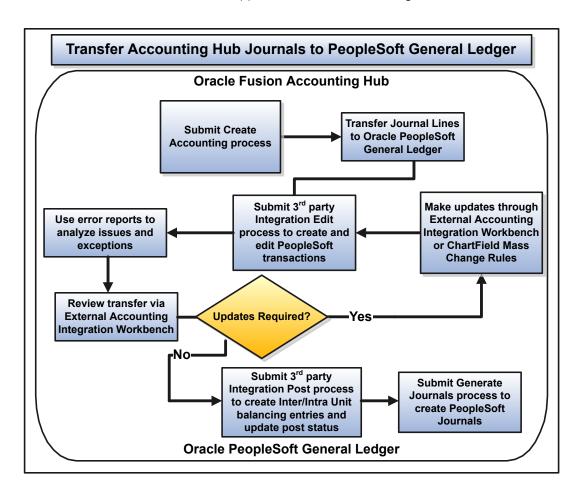
The ChartField Mass Change Rule can be applied once per accounting entry line. The data that is changed by ChartField Mass Change Rules is identified with a different edit status to prevent another ChartField Mass change rule. To reprocess entries changed by ChartField Mass Change Rules, use the option on the Workbench component to reload the transaction from the beginning.

PeopleSoft General Ledger Accounting Process: Explained

Submit the PeopleSoft 3rd Party Integration Edit (GL_FAIJEDIT) accounting process. The process extracts data from Oracle Accounting Hub records, and inserts data into the Accounting Entry Definition record used by the PeopleSoft Journal



Generator in creating PeopleSoft General Ledger journals. According to External System Setup, the process edits the ChartField Combination rules and or applies ChartField Mass Change Rules.



Run the External Accounting Integration Error Report

Any errors noted during the accounting processing are reported to help identify the necessary adjustments. You must provide the External System ID, description, and the business unit or units to include in the External Accounting Integration Error Report.

Run the Combo Edit Error Report

During the 3rd Party Integration Edit Process the Combo Edit validation process can be executed depending on the External System Setup option. You can review the errors when signed in to the Combo Edit Error Report. You must provide the External System ID, description, and the business unit or units to include in the Combo Edit Error Report.

Review Results Using Integration Workbench

Use the Integration Workbench to review the results of the Accounting Hub transfer to PeopleSoft General Ledger.

After processing the accounting entries (3rd Party Integration Edit process), use the External Accounting Integration Workbench inquiry to drill back to Accounting Hub and analyze detail information. Each Accounting Hub journal entry is represented by a separate FAI Transaction ID (Financial Accounting Integration Transaction Identifier).



- Note: Review the Accounting Hub accounting entries through the workbench. Run the 3rd Party Integration Post process or Generate Journals Request in PeopleSoft General Ledger to create journals without reviewing Accounting Hub information through the External Accounting Integration Workbench.
- Enter the required criteria (External System ID and Business Unit) and further limit your results by entering additional criteria. You can search and retrieve the accounting header results and accounting line information that meet the criteria.
- You can also review transactions before running the 3rd Part Integration post process. If any change is required, you
 can also run a manual mass change by using the Mass Change option in the Action field
- Select the Show Detail Values icon in the Accounting Header Results group box to go to the External Accounting Entries page.
- Use the External Accounting Entries page to view the status and details of each Transaction ID.
- Select the Drill to Source icon from the Accounting Line Results group box to go to the Accounting Hub inquiry page.

Changing ChartField Values Using Integration Workbench

Use the Integration Workbench to run a manual mass change for ChartField values.

- Enter the required criteria (External System ID and Business Unit) and further limit your results by entering additional criteria. Set the Action to Mass Change.
- Search and retrieve the resulting accounting header and line information that meet the criteria.
- Enter the ChartField field name and the target field value that are changed by this mass change.
- Select the Run button and confirm that you apply the changes.

Submit the PeopleSoft 3rd Party Integration Post Process

Use the 3rd Party Integration Post run control page to run the 3rd Party Integration Post process (GL_FAIJPOST) to create Inter-Unit and Intra-Unit balancing entries. You can also update the status indicating that the entries are ready to be processed by Journal Generator.

Submit PeopleSoft Post Integration Process: Explained

Use the Third Party Integration Post run control page to launch the Third Party Integration Post process (GL_FAIJPOST) that creates InterUnit and IntraUnit balancing entries. You can update the status indicating that the entries are ready to be processed by the Journal Generator.

The Third Party Integration Post process is edited with the valid accounting entries.

Process Frequency	Select one of the following values when processing the request:
	Process Once - executes the request the next time the batch process runs. After the batch process runs, the process frequency automatically reverts to Don't Run. Always Process are accounted the request area time the batch process runs.
	 Always Process - executes the request every time the batch process runs. Don't Run - ignores the request when the batch process runs
External System ID	Select the user-defined External System ID (External System Setup page) that includes the PeopleSoft Journal Template name to be used by the Journal Generator and information related to



	processing options and Drillback URL for access to view the entries in Accounting Hub. System ID corresponds to any 3rd party System that integrates with PeopleSoft GL. This setup is required to map fields from 3rd party system to PeopleSoft GL (such as: business unit, ledger, currency, ChartField values, and so on) and the field that identifies the Journal Header Transaction ID.
From Accounting Date To Accounting Date	Select the range of dates to be used for selecting accounting entries for processing.
Business Unit	Select the business unit or units to be processed for this request. At least one business unit is required

Processing InterUnit IntraUnit Entries

All InterUnit and IntraUnit processing for Accounting Hub entries occur within the PeopleSoft application to properly create the balancing entries for PeopleSoft InterUnit and IntraUnit journals.

The InterUnit and IntraUnit balancing entries are created by the centralized PeopleSoft InterUnit process (IU_PROCESSOR) after the entries have been edited, processed, modified, and reviewed. The anchor entity is defined according to the anchor flag value from the accounting entry line, and the InterUnit balancing entries are created from that information.

M

Note: All prerequisite PeopleSoft InterUnit and IntraUnit setup must be complete before running this integration. For example, Overall Installation Options (InterUnit fields), InterUnit, and IntraUnit templates are configured for the wanted ChartField values, and General Ledger Definition - Inter/IntraUnit page setup.

Submit Generate Journals Request: Explained

The generate journals process applies the accounting definition to Oracle Accounting Hub journals to create PeopleSoft General Ledger journals. You can configure the process to run at regular intervals. However, you can review the transferred Oracle Accounting Entries prior to generating the PeopleSoft General Ledger journals.

Field	Descriptions
Journal Processing Options	This can be left blank for generating journals from transferred Accounting Hub information. Optionally, you can check Edit and Post options.
Process Frequency	Select one of the following values when processing the request:
	Don't Run - ignores the request when the batch process runs.
	 Always - executes the request every time the batch process runs.
	 Once - executes the request the next time the batch process runs. After the batch process runs, the process frequency automatically reverts to Don't Run.
SetID	Enter the SetID for which the delivered Accounting Definition Name, FAIJRNL, is defined. If the SetID value is not SHARE in this case, you are unable to select the correct Accounting Definition Name since that is the SetID for the delivered value.
Accounting Definition Name	For Oracle Accounting Hub integration, select "FAIJRNL" (the delivered accounting entry definition).



Field	Descriptions			
Application Business Unit	Select an application business unit that the application uses to derive the prompt values for selection. Journal Generator uses the system source from the Accounting Entry Template to derive the prompt table for the application business unit defined for the request.			
Ledger Group	Select the ledger group that includes the ledgers that you would like to process. You can leave this field blank to process all ledger groups.			
Template	Journal Generator creates journal entries for the specified journal generator template. The SetID of the prompt values is derived from the SetID entered earlier on the page.			
From Date Option	Select a value to further define which accounting entries Journal Generator extracts with this request. Journal Generator compares these dates to the accounting dates associated with each accounting entry. Select from the following values:			
	 Specify Date: Uses the date specified to identify the period. Current Date: Uses the date on which Journal Generator is run. Begin Date - From Period: Uses the beginning date of the From Period in the ledger. Begin Date - To Period: Uses the beginning date of the To Period in the ledger. No From Date: Uses all open dates of the ledger before the chosen To Date. This means that the application processes all dates up to the date in the To Date field. Process Date: Date specified on the business unit journal process date option. It can be a 			
From Date	specified date defined on the business unit, or it can be the current date. Enter the begin date if you have selected the Specify Date option.			
TIOH Date	Litter the begin date if you have selected the specify Date Option.			
To Date Option	Select a value to further define which accounting entries Journal Generator extracts with this request. Journal Generator compares these dates to the accounting dates associated with each accounting entry. Select from the following values:			
	 Specify Date: Uses the date specified to identify the period. Current Date: Uses the date on which Journal Generator is run. End Date - From Period: Uses the ending date of the From Period in the ledger. End Date - To Period: Uses the ending date of the To Period in the ledger. Process Date: Date specified on the business unit journal process date option. It can be a specified date defined on the business unit, or it can be the current date. 			
To Date	Enter an end-of-range date in this field if you selected the Specify Date option.			

Create ChartField Mass Change Rule Definition

If the Accounting Hub journal is rejected by PeopleSoft General Ledger due to combo edit validation, you can update the ChartField values with a ChartField Mass Change. The first step to making such changes is to define a ChartField Mass Change Rule.

There are two parts to defining a ChartField Mass Change Rule. First, define the criteria under which the rule is to be applied. This is done by specifying which ChartField values are selected for update. Second, define the target values, which are the values which should be used to replace the selected ChartFields. These rules are applied when the ChartField Mass Change process is enabled on the External System user interface.

 Update the ChartField values. You can't update the header information, such as the accounting date, and line amounts can't be updated.



• You can update the Oracle Accounting Hub journals by PeopleSoft rules only once. Once you have updated, no additional revisions are possible.

Field	Descriptions
Set ID	The SetID value is used as the Set Control Value field to drive the possible ChartField values for the criteria and target prompt tables. The Record Group FS_70: ChartField Mass Change Rules Definition / Rule Groups drives the rule ID prompt to show only the rules that can be applied to the selected SetID.
Rule Name	Define a name for the Mass Change Rule
Description	Provide a description for the rule.
Effective Date	Define the date when this rule becomes valid. This date is compared to the accounting date to determine whether the rule is effective.
Status	Select one of the following options:
	 Active - The rule is actively applied if the effective date is current.
	 Inactive - The rule is not active even if the effective date is current.
Rule Criteria	The Rule Criteria can be defined using one or more fields from the Journal Generator record with a condition and a value. At least one criteria and criteria value must be defined.
Field Name	Select any field from the Journal Generator record to which you want to apply criteria for processing the journal lines that are to be updated. This is typically a ChartField, but can also be the accounting date or other fields mapped as part of the External System Setup.
Operator	The possible values are:
	• equal
	• not equal
	• greater than
	• less than
	The operator is used to evaluate the field value to determine whether to apply the rule for selection
Field Value	Select from the list of values to apply the rule. The prompt table values are driven by the selected SetID.
Target Values	Select any ChartField value or a combination of ChartField values to be changed by the rules. At least one target field and target value should be defined.
Field Name	Select one ChartField or more ChartFields to be updated by this rule. You can select more than one ChartField.
Blank Value	Select this check box to update the blank values for a selected ChartField.
Field Value	Indicate the wanted values for the ChartField. You must specify a value for each row.



Create ChartField Mass Change Rule Group

Once you have defined the ChartField Mass Change Rules necessary to accomplish the ChartField value changes, you define a ChartField Mass Change Rule Group to combine these rules and identify the processing sequence they must be applied.

Field	Descriptions
SetID	The SetID value is used as the Set Control Value field to drive the possible ChartField values for the criteria and target prompt tables.
ChartField Mass Change Group	Define a name for the Mass Change Rule Group. You can select the group on 3rd Party Integration Edit request page.
Effective Date	Define the date when this rule becomes valid. This date is compared to the accounting date to determine whether the rule group is effective.
Status	Select one of the following options:
	 Active - The rule group is actively applied if the effective date is current. Inactive - The rule group is not active.
	Inactive - The rule group is not active.
Description	Provide the description for the ChartField Mass Change Rule Group.
Sequence	Determines the order in which the ChartField Mass Change Rule is to accounting entries. Only one ChartField mass change rule is applied per accounting line.
ChartField Mass Change Rule	Select the ChartField Mass Change Rule to be included in the group and applied to accounting entries.

FAQs for Define Applications Integration with PeopleSoft

When do I review the Journal Generator Template?

The Journal Generator Template determines how the Oracle PeopleSoft (PeopleSoft) Journal Generator creates and summarizes journals. PeopleSoft delivers a sample template entitled **FAH_PSFT**, and contains all the required configurations. Optionally, you can review and update the template to change options such as the journal mask, date option, summarization, or other predetermined selections.

Note: You can assign the meaningful journal source and journal ID mask to easily identify the journals that are created from Oracle Fusion Accounting Hub entries.





7 Define Accounting Transformation

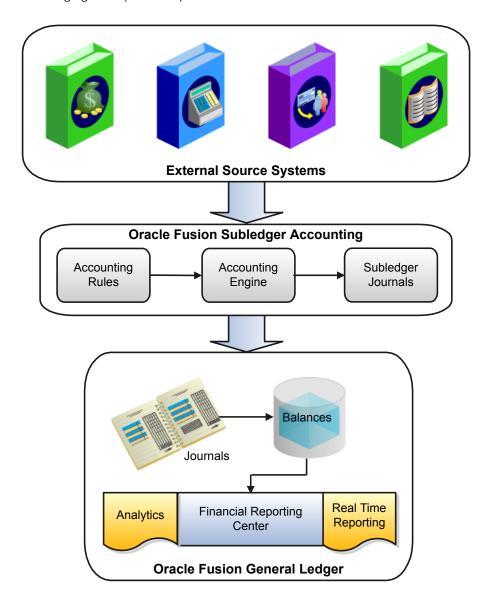
Accounting Transformation Configuration: Overview

Accounting Transformations: Overview

Oracle Fusion Accounting Hub creates detailed, auditable journal entries for source system transactions. The subledger journal entries are transferred to the Oracle Fusion General Ledger. These general ledger journals are posted and update



the general ledger balances. Then the balances are used by the Financial Reporting Center for reporting and analysis. The following figure depicts this process.



Accounting transformations refer to the process of converting transactions or activities, referred to as accounting events, from source systems into journal entries. Source systems may be diverse applications that have been purchased from non-Oracle software providers or created internally. Often, source systems are industry-specific solutions. Examples of source systems are core banking applications, insurance policy administration systems, telecommunications billing systems, and point of sales systems.

Accounting Transformation Flow Chart

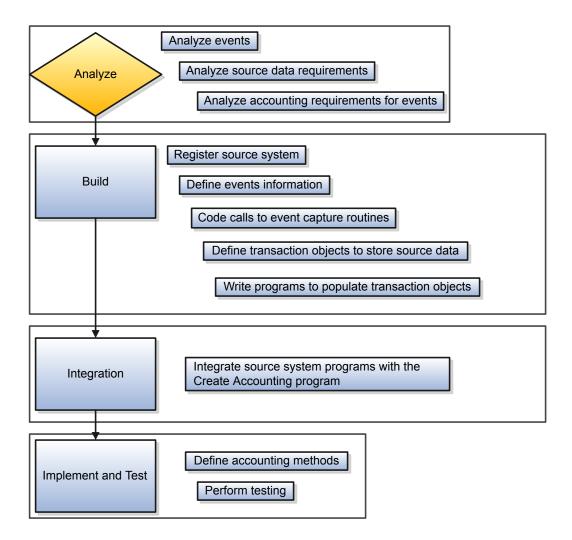
When using the accounting transformation implementation process:

• Start with an analysis of your current system.



- Determine which source systems have transactions or activities that must be accounted and reported through the Oracle Fusion Accounting Hub.
- Register the source system transactions and activities which have financial impact in the Accounting Hub to make them eligible for accounting.
- Create accounting rules that indicate how each of the accounting events is accounted.
- Group these rules and assign them to ledgers to create a complete definition of the accounting treatment for the transactions and activities from the source system.

The following figure summarizes the accounting transformation process.



Accounting Transformation Steps

Complete the steps described in the following table in the order listed to account for accounting events of each of your source systems.



Implementation Phase	Step Number	Description
Analysis	1	Analyze source system transactions or activities to determine what accounting events to capture.
	2	Analyze transaction objects requirements.
		 What source transaction information is available and needed for accounting? What reference information is needed
		for reconciliation and reporting?
	3	Analyze and map the source system's current accounting.
Definition and Build	4	Register source systems and define event model, including: process categories, event classes, and event types.
	5	Code calls to event capture routines.
	6	Build programs to extract the information from the source systems and populate it in the Accounting Hub transaction objects.
	7	Run the Create and Assign Sources program and revise source definitions and map accounting attributes.
Integration	8	Create programs that capture accounting events and their related information and send it to the Accounting Hub.
Implement and Test	9	Create accounting rules.
	10	Perform comprehensive testing to ensure that all accounting is correctly generated.

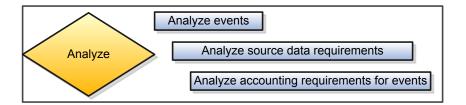
Accounting Transformations Analysis: Explained

The analysis phase of accounting transformation implementation includes three steps. The steps are:

- Analyze Accounting Events
- Analyze Source Data Requirements



Analyze Accounting Requirements for Events



Analyze Accounting Events

Some business events have financial accounting significance and require the recording of financial information. These business events are known as accounting events and provide the data used in accounting transformations.

Examples of business events from a revenue recognition or billing system include:

- Complete an invoice
- Record a payment
- Record late charges

Examples of business events from a point of sale system include:

- Record an order
- Accept a payment
- Receive a return

Examples of business events from a loan (core banking) system include:

- Originate a loan
- Fund a loan
- Record late charges for a loan

An accounting event and its associated transaction data typically relate to a single document or transaction. However, the nature of source systems may prevent them from extracting this discrete information and sending it to Oracle Fusion Accounting Hub for processing. In some cases, summarized event information, such as overall customer activity for the day, is sent for accounting transformation.

The first task is to carry out a complete analysis to determine which accounting events are captured. This analysis incorporates both the functional requirements for accounting for the source system events. It also includes a review of how the events can be captured. There may be limitations on the source system, as well as volume considerations that make it desirable to capture summarized event information such as total customer activity for a day.

Complete the following analysis to identify accounting events:

- Identify the life cycle of documents or business objects and the transactions that affect their status.
- Identify events in the life cycle that may have financial implications to ensure that they are captured and accounted.
- Identify all business events for which contextual data and transaction amounts are available.

While it's not required, this provides maximum flexibility for defining accounting transformation rules.



Analyze Source Data Requirements

Verify that all the required sources, such as accounting amount and date, that can potentially be used to create subledger journal entries are included in the accounting transaction objects. Sources are the appropriate contextual and reference data of transactions. They provide the information that is used to create subledger journal entries. For example, the following items could be used:

- Transaction amount as the debit amount
- Transaction date as the accounting date
- Customer account ID as part of the description

Complete the analysis to determine what source data is necessary to successfully create subledger journal entries from transactions.

Flexibility in creating accounting rules is dependent on the number of sources available. Consider the balance between providing all the information that can be extracted versus how much is practical to send based on your processing resources. The following list provides examples of source data:

- Amounts including entered, accounted, and conversion data
- Dates
- Descriptions
- Accounts
- Customer information
- Transaction type information

Study the transaction objects data model used by the Oracle Fusion Accounting Hub. The data model provides detailed information about the different types of transaction objects. Transaction objects are the views and tables that store transaction data in the standardized form required by the Create Accounting process.

When specifying optional header and line objects, use single table views. If you specify optional objects as multitable views, it can result in poor performance.

Data stored in transaction objects must also satisfy accounting transaction objects validation rules. These rules verify both completion and validity of the data.

Analyze Accounting Requirements for Events

Some source systems may already produce accounting entries, while others may produce raw transactions with no associated accounting. As part of the analysis, determine how much transformation is required to produce subledger journal entries. Once this is done, examine the subcomponents of the journal entry rule set to determine how to complete rules to produce the required subledger journal entries. This exercise helps determine which subledger journal entry rule set subcomponents must be defined for the source systems data to be properly transformed into subledger journal entries.

Journal entry rule set subcomponents include the description rule, account rules, journal line rules, and supporting references.

Such an analysis should, at a minimum, answer the following questions:

- Under what conditions is each of the lines in the subledger journal entry created?
- What is the line type, debit or credit, of each subledger journal entry line?
- What description is used for the subledger journal entry?
- How are the accounts derived for the entry?
- What information may be useful for reconciling the subledger journal entry to the source system?

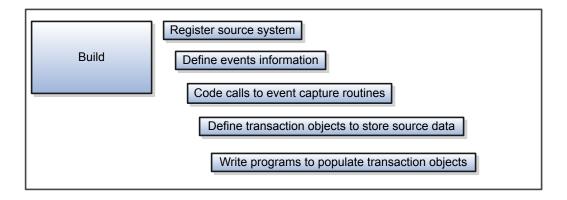




Caution: This list is not comprehensive.

Accounting Transformations Definition and Build: Explained

This section describes the steps for the accounting transformation definition and build phase of the Oracle Fusion Accounting Hub implementation.



Register the Source System and Define Events Information

After registering the source system, set up the accounting event model. The accounting events from the source system are registered in the Oracle Fusion Accounting Hub.

Define Process Categories

 Process categories group event classes, and can be used to restrict the events selected for accounting when users submit the Create Accounting process. Selecting a process category when submitting the Create Accounting process indicates that all associated event classes and their assigned event types are selected for processing. This may be useful for segmenting events due to processing volumes.

Define Event Classes

- Event classes represent transaction types and are used to group event types. For example, when accounting for transactions from a banking system, group the event types Loan Origination, Loan Scheduled Payment, and Loan Late Payments into an accounting event class for Loans.
- For each event class, register the transaction objects that hold source data for the event types in that class.

Define Event Types

For each transaction type, specify each business event that can have an accounting impact.

Code Calls to the Accounting Event Capture Routines

Using application programming interfaces (APIs), create programs to capture the accounting events. The Create Accounting process combines the event information with the transaction object information and the accounting rules to create subledger journal entries. The Create Accounting process reads the event type for each event. Based upon the event type and the primary ledger, it determines which set of accounting rules should be applied to create the subledger journal entry. Once it



determines which rules to use, it gets the information from the event and the transaction object rows related to the event to create the journal entry.

The following APIs for creating and updating accounting events are provided:

- · Get Event Information APIs to get event information related to a document or a specific event
- · Create Event APIs to create accounting events, individually or in bulk
- Update Event APIs to update events and keep them consistent with related transaction data
- Delete Event APIs to delete events

Define Transaction Objects and Write Programs

Transaction objects are tables or views defined for each event class, capturing source transaction data for accounting events. The Create Accounting process gets the source transaction data from the transaction objects to generate journal entries.

Different types of transaction objects exist, indicating whether they are used at the header or line level, and whether they hold translated values:

- Transaction object headers for header level sources that are not translated
- Transaction object headers Multi-Language Support (MLS) for translated header level sources
- Transaction object lines for line level source values that are not translated
- Transaction object lines MLS for translated line level sources

Header sources have the same value for all transaction lines or distributions associated with an accounting event. These sources are typically associated with a transaction header or with transaction reference data. An example of a header standard source for a mortgage loan is the loan number. A mortgage loan can have only one loan number. This number would be on the header transaction object and would not vary by line number.

Line sources have values that can vary by the transaction lines or distributions associated with an accounting event. They must be stored in the transaction objects at the line level.

To create and register transaction objects, perform the following tasks:

- Create transaction objects under FUSION schema to store the source transaction data.
- Write programs that populate the transaction objects with source values for each accounting event.
- Assign transaction objects to event classes in the user interface.

Not all transaction objects must be populated for each accounting event. Consider the following points:

- Transaction objects can be required or optional; at least one header transaction object is required.
- For each event class, define at least one header transaction object as always populated.
- Header transaction objects cannot have more than one row for each event and ledger combination.
- If line transaction objects are assigned to an event class, define at least one of them as always populated.
- If MLS header or MLS line transaction objects are assigned to an event class, it is not necessary for any of them to be always populated. However, there must be at least one corresponding header or line transaction object assigned to the same event class.
- When creating optional transaction objects specify them as single table views. Specifying optional objects as multitable views may result in poor performance.

Accounting event classes can also share transaction objects. For example, when accounting for a core banking system, use the same transaction objects line table or view for both of the event classes: Fixed Rate Mortgages and Variable Rate Mortgages.



Transaction objects must be populated before the accounting for the events occurs. Otherwise, the source transaction information is not available to generate the journal entries for the events.

Transaction objects can be populated in advance of running the Create Accounting process. They can also be populated as part of the Create Accounting process by customizing **xla_acct_hooks_pkg** to automate this coordination.

Accessing the Transaction Objects

Transaction objects should be created under the **FUSION** schema. Then, select, insert, update and delete access should be granted to **FUSION_RUNTIME** for all the transaction view and objects.

A sample grant command is: Grant select, insert, update, delete on fusion.xxfah_trx_headers_v to fusion_runtime; if xxfah_trx_headers_v is created as a transaction object.

Create Sources

After the transaction objects are registered, sources and source assignments to event classes are created based on these objects. Assigning sources to event classes makes them available for creating accounting rules for those classes. The transaction objects column names are used to generate sources. Each column in each transaction object is registered as a separate source. These sources are used in the definition of accounting rules used in creating journal entries.

Create and Assign Sources process also validates the transaction objects by verifying that.

- All transaction objects, defined for the event class, exist in the database.
- All transaction objects, based on the transaction object type, contain the appropriate primary key columns of the correct data type.
- The syntax of all join conditions between the transaction and reference objects is correct.
- A reference object is not registered multiple times for an event class.
- A reference object is not assigned to another reference object.
- Existing sources or source assignments created from previous run of this process continue to be consistent with the transaction objects.
- Any existing accounting attributes mapping previously done continue to be consistent with the transaction objects and accounting attribute definitions.

Revise Source Definitions and Assign Accounting Attributes

Once sources have been created, revise the source definitions before they can be used. These revisions are:

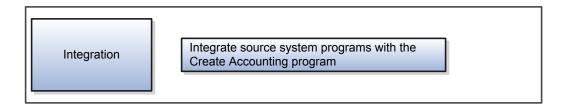
- Source names are the same as the transaction object column names. These can be revised to be more business user-friendly so they are easily understood when configuring accounting rules.
- Sources that correspond to accounting flexfield identifiers are marked as accounting flexfield.
- Whenever appropriate, sources can have lookup types or value sets assigned. Assigning a value set or lookup type
 enables you to predefine valid values for the source that is used to create accounting rules.

After the sources are created, they must be mapped to the accounting attributes for each event class. An accounting attribute is a piece of the journal entry. The mapping of sources to accounting attributes specifies how the Create Accounting process gets the value for each piece of the journal entry. For example, an attribute of entered currency is used to map source values to the entered currency field for subledger journal entry lines.



Accounting Transformations Create Accounting Process: Explained

This section describes the accounting transformation implementation steps for the integration of source system programs with the Create Accounting process in the Oracle Fusion Accounting Hub.



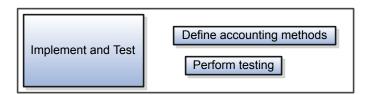
Integrate Source System Programs with the Create Accounting Process

For Oracle Fusion Subledger Application, integrate source system programs to create accounting events using application programming interfaces (APIs).

For Oracle Fusion Accounting Hub, you must customize the xla acct hooks pkg.

Accounting Transformations Implement and Test: Explained

This section describes the steps in the accounting transformations implement and test phase of the Oracle Fusion Accounting Hub implementation.



Accounting methods group subledger journal entry rule sets together to define a consistent accounting treatment for each of the accounting event classes and accounting event types for all source systems.

The following steps must be completed:

- Define accounting methods
- Perform testing

Define Accounting Methods

Define accounting methods to group subledger journal entry rule sets to determine how the source system transactions are accounted for a given ledger. Your goals in defining accounting methods are to:

- Ensure regulatory compliance
- Facilitate corporate financial reporting



- Enable audits
- Facilitate reconciliation to source systems

Assign journal entry rule sets to event class and event type combinations in an accounting method to determine how the subledger journal entries for that class or type are created.

The following are the subcomponents of a journal entry rule set:

- Journal Line Rules: Determine basic information about a subledger journal entry line. Such information includes whether the line is a debit or credit, the accounting class, and if matching lines are merged.
- Description Rules: Determine the descriptions that are included on subledger journal entry headers and lines. Include constant and source values in descriptions.
- Account Rules: Determine which account should be used for a subledger journal entry line.
- Supporting References: Optionally used to store additional source information about a subledger journal entry and are useful for reconciliation of accounting to source systems.

You can attach conditions to journal line rules, description rules, and account rules components. A condition combines constants, source values, and operands to indicate when a particular journal line rule, description, or account rule is used. For example, for mortgage loans, you can elect to use a specific loan receivable account based on the loan type.

Perform Testing

Once the setup is complete, testing should be comprehensive to ensure that all accounting is correctly generated. To complete testing, use accounting events and information from the source system, that is populate the transaction objects. This should, at a minimum, include testing that:

- Accounting events are successfully created.
- Sources are available for creating subledger journal entries.
- Subledger journal entries accurately reflect the accounting rules. Test that:
 - Subledger journal entries contain the appropriate dates, amounts, descriptions, and accounts.
 - Conditions used to determine journal line rules, account rules, or descriptions are valid.
- Entries are successfully summarized when transferred to the general ledger.
- Subledger journal entries are successfully transferred and posted to general ledger.
- Note: The above list is not intended to be comprehensive.

Analyze Accounting Events

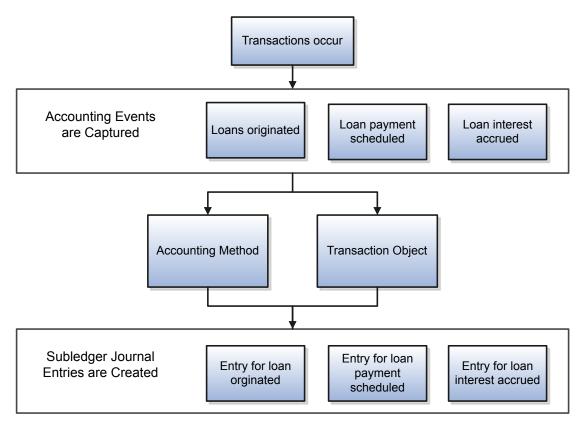
Accounting Event Analysis: Overview

Accounting events have financial accounting significance and are used as a basis for recording financial information.



This diagram depicts the process to create subledger journal entries from accounting events. In this case we are using a custom Loans application as an example. Further explanation in included in the succeeding text.

Creating Subledger Journal Entries from Accounting Events for Loans Custom Application



As illustrated in the above diagram, after transactions occur, accounting events are captured to record their accounting impact. Accounting events can be captured as transactions are generated in the source system. They can also be captured in bulk as part of a standard daily close or batch process. When accounting events are captured is based upon the flexibility of the source system and the frequency with which accounting should be created. For each eligible event, the transaction object contains contextual information about the event, such as source values.

The setups associated with a ledger use the source data for the event to create the appropriate subledger journal entry. Each accounting event can be used to create one or more subledger journal entries. Subsequently, the accounting event links transactions to their corresponding subledger journal entries.

Capturing Accounting Events: Explained

All business events that can have an accounting impact should be captured as accounting events.

Identification and Significance of Accounting Events

The following procedures can assist in the analysis and identification of accounting events:

Identify transactions that may have a financial impact on your organization.



 Identify the life cycles of these transactions and the business events that change the state of the transactions throughout their life cycles.

Business events vary by industry and organization. Examples of business events include the contract, order and delivery of goods and services, and receipts and payments to third parties.

Some business events have financial significance. Their impact must be accounted.

The following examples have a financial accounting impact and therefore are accounting events:

- Originating a loan
- Funding a loan
- Applying payment to a loan

Not all events are accounted. As an example, consider a loan application. The loan origination event results in accounting if the accounting method is Accrual, but may not result in accounting if the accounting method is Cash Basis.

Accounting Event Attributes: Explained

When an accounting event is captured, different event attributes are passed to the Oracle Fusion Accounting Hub.

The following table lists some of the attributes that are stored for an accounting event along with their corresponding descriptions.

Attribute	Description
Event ID	Accounting event internal identifier; provided by Create Accounting.
Event Number	Unique event sequence number within a document. Create Accounting processes events in the order of their event number.
Event Type	Represents the business operation that can be performed on the end-user Transaction Type event class and has accounting significance.
Transaction Identifiers	Identifies the document or transaction in the subledger tables and constitute the primary key of the transaction.
Event Status	Available statuses are: Incomplete The application is not yet ready to account for the event. Unprocessed Finalized accounting has not yet been generated for this event. No Action There is no accounting that must be done for this event. Processed Accounting was done for this event in final mode with no validation failures.
Event Date	Date of the accounting event that originated the journal entry.



Attribute	Description
Transaction Context Values	
	Legal EntityLedgerAsset Book
Application Specific Attributes	Additional columns are provided for implementors to store data drawn from the transaction model (state) at the time the event is captured. These can be useful in cases where the transaction data changes and information are needed on the original event.
Security Context Values	Provide the event's security context. Examples include organization identifier and asset book.
On Hold Flag	Indicates whether there is a gap ahead of an accounting event. If there is a gap, the event is put on hold.
	The Oracle Fusion Accounting Hub does not process accounting events on hold due to a gap.

The Oracle Fusion Accounting Hub event tables store the event data for these attributes. The presence of this data enables the creation of individual subledger journal entries for each accounting event. It also provides an audit trail between the subledger journal entry and the transaction data of the accounting event.

Event capture routines populate these tables when the events are created.

Transaction Context Values: Explained

For the accurate creation of subledger journal entries, an accounting event's transaction context must be passed to the Oracle Fusion Accounting Hub. This context includes the following:

Ledger (Mandatory)

Each accounting setup requires a primary ledger. This primary ledger acts as the main record keeping ledger for one or more legal entities that record and report on all financial transaction using the main:

- Chart of accounts
- Accounting calendar
- Subledger accounting method
- Currency

To maintain an additional accounting representation, use secondary ledgers.

Valuation Method (Optional)

In some cases, the rules for how accounting events should be accounted require that they be subject to differing valuation methods for different ledgers. The Accounting Hub assumes that each subledger transaction and accounting event is for a single valuation method.

Legal Entity (Optional)

Accounting events inherit the legal entity from the transaction it is associated with. Each transaction entity is stamped with a single legal entity.



Event Status: Explained

The event status is an indicator of what actions have been completed on the transaction and what operations are yet to be done.

The Create Accounting process makes updates to this status as the accounting process progresses. Once Create Accounting successfully processes the accounting event, the status of the event is updated to Processed.

Event Status Details

The table below lists the event statuses along with their corresponding details. At any point of time, an event can have only one of these statuses.

Status	Details
Incomplete	The accounting event data is in the process of being created.
	 Some of the accounting event data cannot be created at this point. There can be validations that haven't yet been performed. No subledger journal entry exists for the accounting event.
	 If the Create Accounting process is run, it does not process accounting events with a status of Incomplete. The subledger application updates this event status.
Unprocessed	All of the transaction data for this accounting event has been created and all validations have been performed.
	The subledger journal entry can be created.When the Create Accounting is run, it processes accounting events with a status of unprocessed.
No action	This status is set when creating or updating events using APIs.
	No subledger journal entry is needed for this accounting event.
Processed	All of the transaction data for this accounting event is created according to the accounting rule definition, all validations are performed and, if appropriate, the final subledger journal entry is created.
	 The transaction data associated with the accounting event should not be changed. If the conditions for all the journal line rules assigned to the journal entry rule set for the event class and type are not met, the accounting event status will be updated to Processed without any subledger journal entry. For those transactions where multiple accounting events are allowed, any changes to the transaction data at this time results in a new accounting event. The changed transaction data is tracked under the new accounting event.

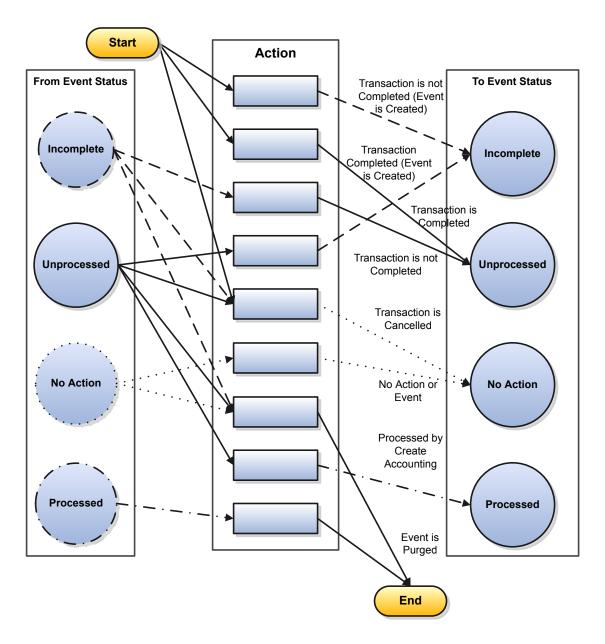
Event Status and the Accounting Event Life Cycle

Every event has a life cycle. The event status indicates what actions have been completed on an event. An accounting event does not necessarily process through each of the statuses.



Accounting Events Life Cycle

Possible event statuses are displayed below.



The above diagram illustrates all the possible status changes for accounting events. An accounting event does not necessarily process through each of the statuses.

The diagram has three blocks:

- The left block, From Event Status, shows the possible initial statuses of an event. These statuses are Incomplete, Unprocessed, No Action, and Processed.
- The center block, Action, represents actions that you complete in subledger applications. These actions result in events being created, processed, or deleted.



• The right block, To Event Status, shows the possible final status based on the action in the central block. The status values include Incomplete, Unprocessed, No Action, and Processed Program.

All of the possible status changes are displayed in the diagram. Not all accounting event types can support all of these status changes. For example, some accounting events, once they have a status of Unprocessed, cannot be updated to an Incomplete status. Implementors are responsible for determining the supported status changes for an event.

There may also be conditions that determine whether the accounting event can move from one status to another. These conditions can vary by accounting event.

For example, a loan that hasn't been processed for accounting may be canceled and is set to a No Action event status. However, if unprocessed loan interest accrual events cannot be canceled, the event status cannot be set to a No Action status.

Event Date: Explained

Each accounting event must have one and only one event date. If there are multiple dates for a particular event type, then one accounting event must be created for each accounting date.

For example, assume a loan is originated and accounted. The next day the loan interest is adjusted. Later, a loan payment is scheduled.

This creates three accounting events:

Accounting Event

Loan Originated (01-JAN-11)

Loan Interest Adjusted (02-JAN-11)

Loan Payment Scheduled (04-JAN-11)

If the transaction has the potential to create multiple events, both the event date and status of the previous event determines whether a new accounting event is created.

Consider the following examples:

- A user cancels the loan before accounting for the loan origination event.
 - old fithe event date for the cancellation is the same as that of the accounting event created for the loan, then the status of the loan origination accounting event is updated to No Action.
 - No accounting event is created for the cancellation.



- A loan origination is canceled after it has been accounted.
 - The accounting event created for the loan origination is not affected by the cancellation.
 - A new event for the cancellation is created and requires processing to create a subledger journal entry.

Accounting Event Order: Explained

All events have an associated accounting event type. For transactions that are document-based, event types correspond to the different operations that can be performed on a document.

Accounting Event Order

If there is an implied order in the accounting events for the different operations on a particular document, then implementors are responsible for enforcing that order. The Oracle Fusion Accounting Hub is neither aware of, nor does it enforce any implied order.

Example: Implied Order in Custom Loan Application

Assume the following three event types for a loan: Loan Originated, Loan Payment Scheduled, and Loan Interest Accrued.

It should not be possible to generate a subledger journal entry to account for a loan payment before the originated event is processed. Similarly, it should not be possible to adjust the loan after it has been canceled. Therefore there is an implied order to these event types.

Correctly Enforcing Implied Order

Assume that the Loan application has four accounting events for a payment as described in the table below:

Event Number	Event Type	Event Date
1	Loan Originated	August 1, 2012
2	Loan Payment Scheduled	August 15, 2012
3	Loan Payment Adjusted	August 16, 2012
4	Loan Payment Canceled	August 28, 2012

In this case, the Loan application has correctly ordered the accounting events. For example, since event processing is by event number, the Accounting Hub does not attempt to account for the adjustment or cancellation of the payment before the Loan Payment Scheduled accounting event has been processed.

The Loan Payment Canceled event must receive an event number greater than the event number of the last Loan Payment Adjusted event type. Similarly, the Loan Payment Adjusted events receive event numbers greater than the Loan Payment Scheduled event but less than the Loan Payment Canceled event.



No Implied Order Between Events of the Same Type

In the following example, assume that event number 3 has an earlier date than event number 2 as described in the table below. Note that event number 2 and event number 3 have the same type of Loan Payment Adjusted.

Event Number	Event Type	Event Date
1	Loan Payment Scheduled	August 1, 2012
2	Loan Payment Adjusted	August 16, 2012
3	Loan Payment Adjusted	August 7, 2012
4	Loan Payment Canceled	August 28, 2012

If you submit the Create Accounting process with a To Date of August 10th, event 1 and then event 3 are processed. Events 2 and 4 remain unprocessed. The latter two events are selected during a later submission of the Create Accounting process with a To Date greater than August 28th. Since there is no implied order between the two Loan Payment Adjusted accounting events, this is acceptable ordering.

If you submit the Create Accounting process with a To Date of the August 31st, all events are processed in event number order, with the August 7th adjustment being processed after the August 16th adjustment. Therefore, the order that the events are processed is affected by the To Date when submitting the Create Accounting process.

Non Enforcing the Implied Order, Incorrect Dating of Events

The tables below describes this example:

Event Number	Event Type	Event Date
1	Loan Payment Scheduled	August 1, 2012
2	Loan Payment Adjusted	August 15, 2012
3	Loan Payment Adjusted	August 16, 2012
4	Loan Payment Canceled	July 20, 2012

In this example, the events are appropriately numbered. If the user specified a To Date equal to or later than August 16th, the events would still be processed in the correct order.

However, the events are not appropriately dated. If the user specified a To Date less than August 16th, the Create Accounting process would be unaware of the Loan Payment Adjusted event number 3, and the Loan Payment Canceled event on July 20th would be processed before the adjusted event.

In this example, the Loan application has incorrectly ordered the accounting events using the event dates. Therefore, it is possible to account for the cancellation of a loan payment before the Loan Payment accounting event has been processed.



Audit Trail and Business Event Restrictions: Explained

Accounting event parameters enable an audit trail between the subledger journal entry and the transaction data for the accounting event.

If any change occurs on a transaction after accounting is created in final mode, such as cancellation of the loan after Loan Origination accounting event is processed in final mode, you can create an accounting event for cancellation or reversal to undo the accounting impact.

You can change the transaction data if its associated event is unprocessed. Business events should achieve the same results by reversing accounting created by the previous accounting events.

Transactions with Multiple Events: Explained

Guidelines for creating accounting events for transactions and documents with multiple events are as follows:

• If a previous event with the same accounting date has a status of Unprocessed, merge accounting events within the same document.

There should be at most one accounting event for the same document and same accounting date that does not have the status Processed. One accounting event can include the data for multiple transactions. This enables accounting at one time for all transaction changes of the same accounting event type.

For example, an unprocessed loan origination is adjusted by adding a line using the same accounting date. When a loan origination hasn't been accounted it has an accounting event with an Unprocessed status. Both the original and created lines must be included on the same accounting event.

• If previous events for a document have a status of Processed, and there are additional adjustments or cancellations applied, an additional accounting event should be created.

For example, consider a loan origination which has been previously accounted. If you cancel the loan origination, an additional accounting event must be created to record the cancellation.

Register Source Systems

Register Source Systems: Critical Choices

Subledger applications can support third-party control account type and calculate reporting currency amounts. If the subledger application is configured to calculate the reporting currency amount, you do not need to provide reporting currency information in the transaction objects



Considerations

The following are considerations when creating a subledger application:

- 1. Determine the subledgers requirement. For example, how many subledgers are to be created? This may depend on what security your company wants to have over its accounting rules.
 - Using the same subledger enables you to share subledger accounting rules, and lets you report across all data easily.
 - Using separate subledgers provides more security across applications and less data in each process run providing better performance. Specific benefits are:
 - If you run two Create Accounting processes at the same time for different applications, they are much
 less likely to contend for database resources. The processes perform better, as the indexes are tuned
 for running with different applications instead of running for different process categories within the same
 application.
 - Enables you to efficiently process different sets of data (different applications) at different times during the day instead of running it as one process.
- 2. Determine the transaction objects requirements. These requirements determine what source data is required to create subledger journal entries from transactions that are captured in transaction objects and shared in reference objects.
- 3. Analyze accounting events to determine what events to capture for the subledger application.

Create programs to capture accounting events using APIs (application programming interfaces) that are provided as follows:

- o Get Event Information APIs to get event information related to a document or a specific event.
- Create Event APIs to create accounting events, individually or in bulk.
- Update Event APIs to update events and keep them consistent with related transaction data.
- Delete Event APIs to delete events.
- **4.** Determine how often to capture accounting events, populate transaction objects, and run the Create Accounting process. This may depend on the immediacy and volumes of accounting requirements in your company.

Accounting Event Model: Explained

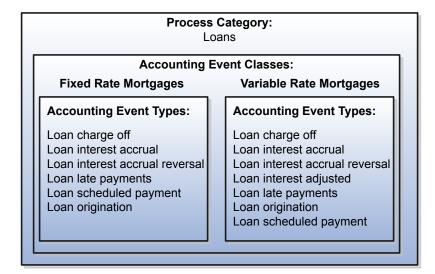
Accounting events represent transactions that may have financial significance, for example, issuing a loan and disposing of an asset. Financial accounting information can be recorded for these events and accounted by the Create Accounting process. When you define accounting events, determine from a business perspective which activities or transactions that occur in your source system may create a financial impact.

Events with significantly different fiscal or operational implications are classified into different accounting event types. Event types are categorized into accounting event classes. Accounting definitions in the Oracle Fusion Accounting Hub are based on event types. An event type must be unique within an application, process category, and event class.

Events are captured when transactions are committed in the subledgers, or they may be captured during end-of-day or period-end processing. For example, a loan is originated, possibly adjusted, interest is accrued, and then the loan is paid or canceled. The accounting events representing these activities can create one or more subledger journal entries, and subsequently link the originating transaction to its corresponding journal entries.



The following is an example of an accounting event model for a loan application:



Process Categories

A process category consists of specific event classes and the event types within those classes. To restrict the events selected for accounting, users can select a process category when they submit the Create Accounting process.

Event Classes

You can assign a transaction view and system transaction identifiers for an event class. Optionally, you can also assign user transaction identifiers and processing predecessors. The transaction view should include all columns that have been mapped to system transaction identifiers for the accounting event class. This view should also include the user transaction identifiers.

System Transaction Identifiers

System transaction identifiers provide a link between an accounting event and its associated transaction or document. An identifier is the primary key of the underlying subledger transaction. This is usually the name of the surrogate key column on the transaction (header), associated with the accounting event. At least one system transaction identifier must be defined for the accounting event class.

When an accounting event is captured, system transaction identifiers, along with other required event data, are validated for completeness.

User Transaction Identifiers

User transaction identifiers constitute the user-oriented key of the underlying subledger transaction, and are typically drawn from one or more database tables. These identifiers are primarily used in accounting events inquiry and on accounting event reports, to uniquely identify transactions. You can specify up to ten columns from the transaction views that are available for inquiry and reports.

The transaction data that identifies the transaction varies by accounting event class. Accounting event reports and inquiries display the transaction identifiers and their labels appropriate for the corresponding event class. The user transaction identifiers can be displayed for an event regardless of its status. This includes the case where the accounting event has not been used to create subledger journal entries due to an error. Or the cases where it has not been processed. The user transaction identifier values are displayed at the time that the accounting event reports and inquiries are run. If a transaction



identifier value has changed after the accounting event was captured, the accounting event reports and inquiries reflect the change.

Processing Predecessors

The processing predecessor establishes an order in which the Create Accounting process processes events selected for accounting.

Event Types

For accounting event types, specify whether their accounting events have accounting or tax impact. When the Create Accounting process is submitted, it only accounts business events that are enabled for accounting.

Transaction Objects: Points to Consider

You may assign transaction and reference objects for each accounting event class in the subledger application. Sources are generated based on the transaction objects and are assigned to the corresponding accounting event classes.

Sources are used to create accounting rules. Subledgers pass information to the application by populating transaction object tables. The columns in these tables are named after the source codes. Transaction and reference objects hold transaction information that is useful when creating journal entry rules for accounting. Transaction and reference objects are defined for an accounting event class. Source assignments to the accounting event class are generated using these objects.

Transaction Objects

Transaction objects refer to the tables or views from which the Create Accounting process takes the source values to create subledger journal entries. Source values, along with accounting event identifiers, are stored in the transaction objects. The Create Accounting process uses this information to create subledger journal entries.

You have several options. You can:

- Create new tables as the transaction objects and create a program to populate them.
- Use views of your transaction data as the transaction objects.
- Use your transaction data tables as the transaction objects.

The transaction objects and or views must be accessible to the Create Accounting process. Typically, an ETL (extract, transformation, and load) program is used to take values from the source system and load them into the database used by the Create Accounting process. The ETL process is done outside of the Create Accounting process.

The following are transaction object types:

- Header transaction objects
 - Implementors must provide at least one header transaction object for each accounting event class. Header transaction objects store one row with untranslated header source values for each accounting event. The primary key of a header transaction object is the event identifier.



Transaction details that are not translated, and whose values do not vary by transaction line or distribution, should normally be stored in header transaction objects. Examples of sources normally stored in header transaction objects include the Loan Number for a loan or the Contract Number for a contract.

- Line transaction objects
 - Line transaction objects are relevant when there are details for the accounting event that vary based upon transaction attributes. For example, a mortgage transaction for loan origination may have multiple amounts, each related to different components of the loan. There may be a loan origination amount, closing cost amounts, and escrow amounts. Each of these amounts could be captured as separate lines, along with an indication of the amount type.

Line transaction objects store untranslated line level source values. There should be one row per distribution, identified by a unique line number. The primary key of a line transaction object is the event identifier and transaction object line number. Transaction details that are not translated, and whose values vary by transaction line or distribution, are normally stored in line transaction objects columns. Examples include the Loan Number for a loan payment.

- Multi-Language Support (MLS) transaction objects
 - MLS transaction objects are relevant to applications that support the MLS feature. MLS transaction objects store translated source values. There should be one row per accounting event and language. The primary key of a header MLS transaction object is the event identifier and language. The primary key of a line MLS transaction object is the event identifier, transaction object line number, and language.

Transaction details that are translated, and whose values do not vary by transaction line or distribution, are normally stored in header MLS transaction object columns. Examples include Loan Terms for a commercial loan. Implementors can avoid having to store source values in header MLS transaction objects by using value sets and lookup types.

Transaction details that are translated, and whose values vary by transaction line or distribution, should normally be stored in the transaction object in columns defined in a line MLS transaction object.

Reference Objects

Reference objects are useful for storing information that is used for creating subledger journal entries. This information may not be directly from the source system or may be used for many entries, thus making it redundant. Use reference objects to share sources information across applications.

For example, store customer attributes, such as the customer class or credit rating in a reference object, and then, use it to account for different journal entries in a loan cycle, such as loan origination or interest accrual. Store information, such as bond ratings and terms, and use it to account for entries across the life of bonds, such as interest accruals or bond retirement.

Reference objects can either have a direct relationship to transaction objects (primary reference object), or be related to other reference objects (secondary).

Validating Sources and Source Assignments: Explained

The Create and Assign Subledger Sources process performs the following source and source assignment validations:

• Source assigned to accounting event class is defined in at least one transaction or reference object associated with the accounting event class.



- Data type of source matches the data type of a corresponding column from a transaction or a reference object assigned to an event class.
- Source is assigned to the event class at a level that is consistent with the transaction or reference object type.

Managing Accounting Sources: Critical Choices

Sources are a key component for setting up accounting rules. Sources represent transaction and reference information from source systems. Contextual and reference data of transactions that are set up as sources can be used in accounting rules definitions.

When determining what sources should be available, it's helpful to begin the analysis by considering which information from your systems is accounting in nature.

Examples of sources that are accounting in nature include:

- Transaction accounts
- Transaction currency
- Transaction amounts

Sources that are not always required for accounting rules include items that are related to the transaction for other purposes than accounting.

Examples of information that may not be specifically for accounting, but which may be useful for creating subledger journal entries are:

- Transaction identification numbers such as:
 - Loan number
 - Customer number
 - Billing account number
- Counterparty information
- Transaction dates

Provide a rich library of sources from your source systems for maximum flexibility when creating definitions for subledger journal entries.

Sources are assigned to accounting event classes by submitting the Create and Assign Sources process.

A distinct difference exists between sources and source values:

- Sources represent the transaction attributes used to create accounting rules.
- Source values are used by the Create Accounting process to create subledger journal entries based upon source assignments to accounting rules.

Sources

Sources must be created prior to creating accounting rules. This predefined step must be undertaken before the application can be used to create subledger journal entries.

To set up appropriate subledger journal entry rule sets, users and those implementing must understand the origins, meaning, and context of sources.



Use business-oriented names for sources to allow accountants and analysts to effectively create accounting rules.

- Enables users to easily identify a source.
- Ensures consistency in nomenclature.

Source Values

Source values are stored in transaction objects. They are the actual transaction attribute values from the source system and are used in creation of the journal entries.

Accounting Attribute Assignments: Points to Consider

The Create Accounting process uses the values of sources assigned to accounting attributes plus accounting rules to create subledger journal entries. Almost all accounting attributes have sources assigned at the accounting event class level. Depending on the accounting attribute, the accounting attribute assignment defaults from the accounting event class. It can be overridden on journal line rules or subledger journal entry rule sets.

Once sources are assigned to accounting event classes, they are eligible for assignment to accounting attributes for the same accounting event classes.

The Create Accounting process uses these assignments to copy values from transaction objects to subledger journal entries. For example, you may map the invoice entered currency to the subledger journal entry entered currency.

Each accounting attribute is associated with a level:

- 1. Header: To be used when creating subledger journal entry headers.
- 2. Line: To be used when creating subledger journal entry lines.

The types of accounting attributes values are as follows:

Values that are Subject to Special Processing

You may have values that are subject to special processing or values that are stored in named columns in journal entry headers and lines.

Examples of accounting attributes are Entered Currency Code and Entered Amount.

Values that Control the Action of the Create Accounting Process

You may have values that control the action of the Create Accounting process when processing a specific accounting event or transaction object line.

An example of accounting attributes of this type is Accounting Reversal Indicator.

Minimum Required Accounting Attribute Assignments

To create a valid journal entry you must, at a minimum, set up the following accounting attribute assignments.

- Accounting Date
- Distribution Type
- Entered Amount
- Entered Currency Code
- First Distribution Identifier

The details and descriptions of these attributes are included in the Accounting Attributes section.



Accounting Attributes

Accounting attribute groups are represented in the tables below:

Accounted Amount Overwrite

- The accounted amount overwrite attribute identifies whether the accounted amount calculated by the Create Accounting process should be overwritten by the value of the accounted amount accounting attribute.
- If the source value mapped to Accounted Amount Overwrite is 'Y', then an accounted amount must be provided.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Accounted Amount Overwrite Indicator	Alphanumeric	Line	Event Class and Journal Line Rule	No	Y - Overwrite accounted amount
					N - Not overwrite accounted amount

Accounting Date

- The accounting date attribute is relevant to all applications. The Create Accounting process uses it to derive the
 accounting date of journal entries. Typically, the event date system source is assigned to the accounting date
 attribute.
- The Accrual Reversal GL Date accounting attribute is relevant to applications using the accrual reversal feature.
 Users can assign system and standard date sources to the Accrual Reversal GL Date in the Accounting Attribute
 Assignments page. When the Accrual Reversal GL Date accounting attribute returns a value, the Create Accounting process generates an entry that reverses the accrual entry.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Accounting Date	Date	Header	Event Class and Journal Entry Rule Set	Yes	Should be in open general ledger period
Accrual Reversal GL Date	Date	Header	Event Class and Journal Entry Rule Set	No	Should be later than the accounting date

Accounting Reversal

- Accounting reversal accounting attributes are relevant to applications that want to take advantage of the accounting reversal feature. The Create Accounting process uses them to identify transaction (distributions) whose accounting impact should be reversed.
- For the Create Accounting process to successfully create a line accounting reversal, the accounting reversal indicator, distribution type, and first distribution identifier should always be assigned to sources. The definition of the accounting reversal distribution type and distribution identifiers mirrors the definition of the distribution identifiers.



Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Accounting Reversal Distribution Type	Alphanumeric	Line	Event Class	Yes, if another accounting reversal accounting attribute is assigned.	
Accounting Reversal First Distribution Identifier	Alphanumeric	Line	Event Class	Yes, if another accounting reversal accounting attribute is assigned.	
Accounting Reversal Second Distribution Identifier	Alphanumeric	Line	Event Class	No	
Accounting Reversal Third Distribution Identifier	Alphanumeric	Line	Event Class	No	
Accounting Reversal Fourth Distribution Identifier	Alphanumeric	Line	Event Class	No	
Accounting Reversal Fifth Distribution Identifier	Alphanumeric	Line	Event Class	No	
Accounting Reversal Indicator	Alphanumeric	Line	Event Class	Yes, if another accounting reversal accounting attribute is assigned.	Y - Reverse without creating a replacement line B - Reverse and create a new line as replacement N or Null - Not a reversal
Transaction Accounting Reversal Indicator	Alphanumeric	Header	Event Class	No	Y - Reversal transaction object header N or null - Standard transaction object header

Business Flow

• The business flow accounting attributes are referred to as 'applied to' accounting attributes. If a transaction is applied to a prior transaction in the business flow, the transaction object must populate sources assigned to 'applied to' accounting attributes with sufficient information to allow the Create Accounting process to uniquely identify a transaction object line for a prior event in the business flow.



- When deriving accounting data from a previous event in the business flow, the Create Accounting process searches for a journal entry line for the prior event using a combination of the 'applied to' accounting attributes and the business flow class of both journal entries.
- The Applied to Amount accounting attribute is used to calculate the accounted amount and gain or loss in crosscurrency applications when business flows are implemented. This attribute value is used to calculate the accounted amount when a source is mapped to the Applied to Amount attribute on a journal line type and the entered currency is different than the original currency entered.
- Note: When enabling a business flow to link journal lines in the Journal Line Rule page, certain accounting attribute values are unavailable for source assignment. These do not appear in the Accounting Attributes Assignments window of the same page because they are copied from the related prior journal entry.

Accounting	Data Type	Journal Entry Level	Assignment to Rules	Assignment Validation Rules
Attributes				Required?
Applied to Amount	Number	Line	Event Class and Journal Line Rule	No
Applied to First System Transaction Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.
Applied to Second System Transaction Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No
Applied to Third System Transaction Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No
Applied to Fourth System Transaction Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No
Applied to Distribution Type	Alphanumeric	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.
Applied to First Distribution Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.
Applied to Second Distribution Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No
Applied to Third Distribution Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No



Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Applied to Fourth Distribution Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No	
Applied to Fifth Distribution Identifier	Alphanumeric	Line	Event Class and Journal Line Rule	No	
Applied to Application ID	Number	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.	Must be a valid application ID
Applied to Entity Code	Alphanumeric	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.	Must be a valid Entity for the application selected in Applied to Application ID

Distribution Identifier

- Distribution identifiers accounting attributes are relevant to all applications. The distribution identifier information links subledger transaction distributions to their corresponding journal entry lines. In addition, many of the subledger accounting features, including accounting reversals, rely on the correct definition and storing of distribution identifiers in the line transaction objects.
- The distribution type and first distribution identifiers are always assigned to sources. If a transaction distribution is identified by a composite primary key, additional distribution identifiers are assigned to standard sources, as appropriate.
- Values for the distribution type and distribution identifiers are always stored in accounting transaction objects. The
 combinations of the values of the system transaction identifiers with the values of the distribution identifiers uniquely
 identify a subledger transaction distribution line.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Distribution Type	Alphanumeric	Line	Event Class	Yes	
First Distribution Identifier	Alphanumeric	Line	Event Class	Yes	
Second Distribution Identifier	Alphanumeric	Line	Event Class	No	
Third Distribution Identifier	Alphanumeric	Line	Event Class	No	
Fourth Distribution Identifier	Alphanumeric	Line	Event Class	No	
Fifth Distribution Identifier	Alphanumeric	Line	Event Class	No	



Attributes Assignment to hales Assignment to hale as a second to hale a second	Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
--	--------------------------	-----------	---------------------	---------------------	-------------------------	------------------

Document Sequence

- The document sequence accounting attributes are relevant to applications that use the document sequencing feature to assign sequence numbers to subledger transactions. The Create Accounting process uses them to provide a user link between subledger transactions and their corresponding subledger journal entries.
- Assign all document sequence accounting attributes to sources or do not assign any.
- The Document Sequence Category Code is made available as an Accounting Sequence Numbering control
 attribute.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Subledger Document Sequence Category	Alphanumeric	Header	Event Class	Yes, if another accounting attribute in the same group has assignment.	
Subledger Document Sequence Identifier	Number	Header	Event Class	Yes, if another accounting attribute in the same group has assignment.	
Subledger Document Sequence Value	Number	Header	Event Class	Yes, if another accounting attribute in the same group has assignment.	

Entered Currency

- Entered currency accounting attributes are relevant to all applications. The Create Accounting process uses them to populate the journal entry line entered currency and amounts.
- The entered currency accounting attributes must always be assigned to sources. The sources assigned to the entered currency accounting attributes must always contain a value.
- For event classes that support cross currency transactions, and more than one entered currency and entered currency amount, multiple event class accounting attribute assignments are created.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Entered Currency Code	Alphanumeric	Line	Event Class and Journal Line Rule	Yes	A valid currency code
Entered Amount	Number	Line	Event Class and Journal Line Rule	Yes	



Ledger Currency

- Ledger currency accounting attributes are relevant to all applications that use the Create Accounting process. The Create Accounting process uses them to populate journal entry accounted amounts.
- If a transaction's entered currency is different from the ledger currency, the Create Accounting process copies the conversion date, rate, and rate type to the corresponding journal entry lines. If the entered currency is the same as the ledger currency, the Create Accounting process ignores the conversion type and conversion rate.
- For event classes that support foreign currency transactions, and therefore more than one conversion rate and reporting currency amount, multiple event class accounting attribute assignments are created.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Accounted Amount	Number	Line	Event Class and Journal Line Rule	No	
Conversion Date	Date	Line	Event Class and Journal Line Rule	No	
Conversion Rate	Number	Line	Event Class and Journal Line Rule	No	
Conversion Rate Type	Alphanumeric	Line	Event Class and Journal Line Rule	No	A valid general ledger conversion rate type or User

Tax

- The tax accounting attributes are relevant to applications that uptake the tax initiative. The tax team uses the tax accounting attributes to link subledger transaction tax distributions to their corresponding journal entry lines.
- Oracle Fusion Tax specifies which tax reference values are required in transaction objects and are assigned to standard sources.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Detail Tax Distribution Reference	Number	Line	Event Class	No	
Detail Tax Line Reference	Number	Line	Event Class	No	
Summary Tax Line Reference	Number	Line	Event Class	No	

Third Party

Third-party accounting attributes are relevant to subledger applications that use third-party control accounts. The
third-party accounting attributes link suppliers and customers to their corresponding subledger journal entry lines in
the supplier and customer subledgers.



- For all subledger transactions that represent financial transactions with third parties, all third-party accounting attributes have sources assigned.
- If a transaction line is associated with a customer or supplier, the transaction objects must include values for all sources mapped to third-party accounting attributes for the event class.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Party Identifier	Number	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.	If party type C - Should be a valid customer account If party type is S - Should be a valid supplier identifier
Party Site Identifier	Number	Line	Event Class and Journal Line Rule	Yes, if another accounting attribute in the same group has assignment.	If party type C - Should be a valid customer account If party type is S - Should be a valid supplier identifier
Party Type	Alphanumeric	Line	Event Class	Yes, if another accounting attribute in the same group has assignment.	C for Customer S for Supplier

Exchange Gain Account, Exchange Loss Account

- The Create Accounting process determines whether there is an exchange gain or loss and derives the account combination based on whether the journal line rule is defined.
- If the gain or loss journal line rule is defined, the account rule assigned to the journal line rule is used to determine the gain or loss account to use.
- If the gain or loss journal line rule is not defined, the gain or loss account assigned to the Exchange Gain Account and Exchange Loss Account accounting attributes is used.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Exchange Gain Account	Number	Header	Event Class	No	
Exchange Loss Account	Number	Header	Event Class	No	

Gain or Loss Reference

The Gain or Loss Reference accounting attribute groups entry lines together when calculating exchange gain or loss.
 The accounted debit and accounted credit amounts for lines with the same gain or loss reference are combined. The total of accounted debit and total of accounted credit are compared to calculate the exchange gain or loss.



Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Gain or Loss Reference	Alphanumeric	Line	Event Class	No	

Transfer to GL Indicator

- The Transfer to GL accounting attribute is relevant to applications which create subledger journal entries that are not transferred to the general ledger. The Transfer to GL process uses this accounting attribute to determine whether to transfer subledger journal entries to the general ledger.
- If the Transfer to GL accounting attribute is not assigned to a source, the Transfer to GL process transfers journal entries for the event class to the General Ledger.
- If the Transfer to GL accounting attribute is assigned to a source and the source is not populated, the Transfer to GL
 process transfers journal entries for the event class to the General Ledger.

Accounting Attributes	Data Type	Journal Entry Level	Assignment to Rules	Assignment Required?	Validation Rules
Transfer to GL Indicator	Alphanumeric	Header	Event Class	No	Should be Y or N

Implement Accounting Event Interfaces

Accounting Event Interfaces: Overview

The Oracle Fusion Accounting Hub provides a set of common APIs to capture accounting events. All event operations must be performed through these APIs. By ensuring that event operations are executed through generic APIs, the architecture meets the needs of implementors:

Insulates implementors from changes in the implementation of an API

The presence of the API reduces dependencies between the Accounting Hub and your source systems.

Implementors do not have to know the underlying Accounting Hub data model to capture accounting events. In addition, any implementation changes made by the Accounting Hub have minimum or no impact on implementors.

The Accounting Hub relies on accounting events to indicate that there are activities from source systems that require accounting.

The Create Accounting process selects accounting events based on criteria specified by users. The Create Accounting process does not check for any functional dependencies between transactions or event types.

Note: For each eligible event, the Create Accounting process retrieves source values from the transaction objects. Subledger journal entries are created using both event and source information.



Procedures to Create Event Capture Routines: Explained

Implementors must undertake the following steps to create event capture routines:

- Perform accounting event setups
- Write product specific wrapper routines
- Integrate event APIs with source systems

Perform Event Setups

As a prerequisite, you must register the application.

An additional prerequisite step is to define and register event process categories, accounting event classes, and event types before events can be captured. The event APIs use event information to perform the necessary validations when creating events.

Write Product Specific Cover Routines

In order to reduce dependencies and facilitate maintenance, it is recommended that you write a wrapper routine on top of the Oracle Fusion Accounting Hub APIs. Wrapper routines can encapsulate source system specific logic to perform necessary validations before creating new events. Map source system specific parameters to the API parameters

For example, if you were implementing the Accounting Hub to capture information for a loans application, you could use a package called LOAN_XLA_EVENTS_PKG, which contains all the APIs to implement accounting events in Loans. The code for this package is shown below.

Assume that this example loans application has two event classes Create Loan and Create Payment. To handle accounting events for loan transactions, you could create a procedure create_loan_event() with p_loan_id as an input parameter, instead of using a generic parameters like source_id_int_1. The procedure create_loan_event() calls the appropriate Accounting Hub API to create an event.

Similarly, create create payment event() to handle accounting events for the event class Create Payment.

```
LOAN_XLA_EVENTS_PKG
-- Procedure to create events for creating a loan
PROCEDURE create loan event
(p loan id
,p_event_type
,p event date
,p event status) IS
1 loan source info XLA EVENTS PUB PKG.t event source info;
1_security_context XLA_EVENTS_PUB_PKG.t_security; BEGIN
BEGIN
-- Perform product specific checks
-- Map generic event API parameter to the product specific columns
1 load source info.application id = loan application id;
l loan source info.legal entity id = l legal entity id;
l loan source info.source id int 1 = p loan id;
-- Call XLA API
XLA EVENTS PUB PKG.create event
(p_event_source_info => l_loan_source_info
,p_event_type_code => p_event_type
,p event date => p event date
,p_event_status_code => p_event_status
,p_event_number => NULL
,p reference info => NULL
```



```
,p_valuation_method => NULL
,p_security_context => l_security_context);
...
EXCEPTIONS
...
END:
```

Integrate Event APIs with Source Systems

It is suggested that implementors create an Enterprise Scheduler Service (ESS) process for the event capture wrapper routines. Using an ESS process instead of directly running the routines in the database will provide for the following:

- Enable implementation of security where some users can be given access to create events for specific source systems but not all.
- Go through the proper channels of executing code on Fusion schema. The ESS job will not require the user to have write access to Fusion schema.

Overview of Event APIs

Getting Event Information: Overview

You may must perform certain checks with respect to events. For example, before creating a new accounting event, it's necessary to check whether there is an existing unprocessed event for the same transaction with the same accounting event type and event date.

You may also want to know the status of a particular event or query events already created for the transaction.

To perform these checks, obtain event information for a transaction by doing the following:

- 1. Determine the system transaction identifiers and accounting event class for the transaction.
 - Note: System transaction identifiers identify the transaction on which events are based. Oracle Fusion Accounting Hub uses these identifiers to search the events table and identify all the events that are related to a transaction.
- 2. Call the function <code>Get_Array_Event_Info()</code> with the appropriate transaction parameters. The function returns an array of all events for a transaction.

To obtain all the events created for a particular event type within a transaction, do the following:

- 1. Determine the system transaction identifiers, event class, and event type of the transaction.
- 2. Call the function <code>get_Array_Event_Info()</code> with the appropriate transaction and event type input parameters. The function returns an array of all accounting events for that transaction and event type. Optionally, pass the event class, event date, and event status to further restrict the rows returned.

To get information about a specific event, do the following:

- 1. Determine the event id.
- 2. Call the function <code>Get_Event_Info()</code> with the event_id parameter. This function returns a PL/SQL record containing all information pertaining to that particular event.



Creating Events: Explained

This section describes the following guidelines on creating events using the create event APIs:

- · Creating a single event
- Creating events in bulk

Creating a Single Event

To create a new event:

- 1. Determine the accounting event type, event date, and event status for the new event.
- 2. Call the Oracle Fusion Accounting Hub function Create_Event() with the appropriate input parameters.

The create Event() API creates a single event at a time. The function returns the event id of the created event.

Creating Events in Bulk

Create events in bulk using the API create Bulk Events().

▲ Caution: Don't use this API for existing transactions that already have events associated with them. For performance reasons, bulk event APIs don't perform checks as to whether events for the transaction already exist. Therefore, use this API only to create events for new transactions that do not have any prior events created.

Updating Events: Overview

Update the event to keep the transaction data and related events synchronized.

You can update an event as long as it is not processed. Once an event is accounted in Final status, you cannot update the event or the data associated with it.

Use the following APIs to update your events:

Update Event

This is an overloaded API used to update the status, type, date, number, and reference information of a single event.

Update Event Status

This API updates multiple events to a given status.

Deleting Events: Explained

To delete all Unprocessed events associated with a transaction:

- 1. Determine the transaction source information.
- 2. Call the Oracle Fusion Accounting Hub function Delete Events() with the transaction source information.
- 3. Optionally, specify the accounting event type, event status, or event date, to restrict the events deleted.



If a transaction is deleted, the pelete_Entity()API must be called to delete the row in the XLA_TRANSACTIONS_ENTITY table.

Updating Transaction Numbers: Overview

In some cases, the transaction number of the transaction is changed. This API updates the transaction number on the events in the XLA_TRANSACTION_ENTITIES table so that they are consistent with the transaction number on the transaction.

The API checks the source information for a valid application, legal entity, event process category, and source identifiers. However, no validation is performed on the transaction number.

Details of Event API

Event API Details: Overview

This section describes the event APIs accessible by implementors to perform event operations. The APIs described are generic and available to all applications.

Event APIs have the following characteristics:

- Event APIs don't issue any commits.
- If an API fails, any changes made are rolled back and a standard exception is raised.
- All parameters are read only (IN parameters); there are no OUT parameters.
- The event date is always truncated.

The Oracle Fusion Accounting Hub does not store the time stamp for an event date.

All functions return a single value or record.

The exceptions to this rule are the functions prefixed with a <code>Get_Array</code> string. For example, the function <code>Get_Array_Event_Info()</code> returns an array.

- Input parameters must be passed in a notation without a position, named notation.
- All the APIs called in query mode lock the event record in NOWAIT mode.

Event APIs have the following types of input parameters:

System transaction identifiers

These parameters capture information such as loan_id. This information is stored with each event to later identify the source transaction for the event. You must pass the source identifiers when creating an event.

Transaction security identifiers

Accounting events are subject to the security of the corresponding transaction. Every accounting event is stamped with its related transaction's security context.

Transaction security parameters capture application-specific transaction security information, such as business unit or ledger.

Transaction reference information



The reference parameters enable you to capture any miscellaneous reference or contextual information for an event. This information is optional and no validations are performed against any reference parameters.

The XLA_EVENT_PUB_PKG package contains the following items:

- PL/SQL record and table structures for common parameters
- CONSTANTS for event statuses

Use these constants and structures when passing and reading values to and from the APIs.

Getting Event Information APIs: Examples

This section provides details on the APIs that obtain event information.

1. XLA_EVENTS_PUB_PKG.GET_EVENT_INFO()

This API returns information for a particular event. The event is identified by specifying the transaction and event identifier. The API locks the specified event before returning any event information.

The API checks all source information for valid application, legal entity, event entity, and source identifiers (IDs). It ensures that the required parameters are not passed as null and that the event ID belongs to the same transaction, as the other transaction information being passed.

The API returns a PL/SQL record containing event information.

```
FUNCTION XLA_EVENTS_PUB_PKG.get_event_info
(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)
RETURN xla_events_pub_pkg.t_event_info;
```

2. XLA_EVENTS_PUB_PKG.GET_ARRAY_EVENT_INFO()

This routine returns information for one or more events within a transaction. The calling program specifies the transaction and event selection criteria. Return information contains data on all events that belong to the specified transaction and fall under the given criteria. The API locks the specified events before returning the event information.

The API checks all source information for valid application, legal entity, event process category, and source IDs. It ensures that the required parameters are not passed as null and also validates the accounting event class, event type, and event status. Note that the API truncates the event date.

The API returns an array of event information.

```
FUNCTION XLA_EVENTS_PUB_PKG.get_array_event_info
(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_class_code IN VARCHAR2 DEFAULT NULL
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)
RETURN xla_events_pub_pkg.t_array_event_info;
```



3. XLA_EVENTS_PUB_PKG.GET_EVENT_STATUS()

This API returns the event status for a specified event. The calling program must specify the transaction and event identifier. The API locks the specified event record before returning the status.

The API checks all source information for valid application, legal entity, event process category, and source IDs. This ensures that the required parameters are not null, and the event belongs to the same transaction as the other transaction information being passed.

This API returns an event status. The Oracle Fusion Accounting Hub has defined all event statuses as Constants.

```
FUNCTION XLA_EVENTS_PUB_PKG.get_event_status

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)

RETURN VARCHAR2;
```

4. XLA_EVENTS_PUB_PKG.EVENT_EXISTS()

This API checks whether an event exists for the specified criteria. It returns True if it finds at least one event matching the criteria; otherwise, it returns False. The API locks the event rows before returning a value.

The API checks all source information for valid application, legal entity, event process category, and source IDs. It ensures that the required parameters are not null and also validates the event class, event type, and event status. The API truncates the event date.

The API returns True if an event is found for the specified criteria; otherwise, it returns False.

```
FUNCTION XLA_EVENTS_PUB_PKG.event_exists

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_class_code IN VARCHAR2 DEFAULT NULL
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_event_number IN NUMBER DEFAULT NULL
,p_event_number IN NUMBER DEFAULT NULL
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)
RETURN BOOLEAN:
```

Creating Event APIs: Examples

This section provides details on the API that create events.

XLA_EVENTS_PUB_PKG.CREATE_EVENT()

This API creates a new event.

The API checks all source information for valid application, legal entity, event process category, and source IDs. It ensures that the required parameters are not null and also validates the accounting event type and event status.

No validations are performed against the reference columns and event number. However, if no event number is passed, the routine populates the next highest event number for that transaction. The event date is truncated.

If an event is created successfully, then the function returns its event ID.

```
FUNCTION XLA_EVENTS_PUB_PKG.create_event (p_source_event_info IN xla_events_pub_pkg.t_event_source_info
```



```
,p_event_type_code IN VARCHAR2
,p_event_date IN DATE
,p_event_status_code IN VARCHAR2
,p_event_number IN NUMBER DEFAULT NULL
,p_reference_info IN xla_events_pub_pkg.t_event_reference_info
DEFAULT NULL
,p valuation method IN VARCHAR2
,p_transaction_date IN DATE DEFAULT NULL
,p_security_context IN xla_events_pub_pkg.t_security
RETURN NUMBER;
FUNCTION XLA EVENTS PUB PKG.create event
(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_type_code IN VARCHAR2
,p_event_date IN DATE
,p event status code IN VARCHAR2
,p_event_number IN NUMBER DEFAULT NULL
,p_transaction_date IN DATE DEFAULT NULL
,p reference info IN xla events pub pkg.t event reference info
DEFAULT NULL
,p_valuation_method IN VARCHAR2
,p security context IN xla events pub pkg.t security)
RETURN NUMBER:
```

2. XLA_EVENTS_PUB_PKG.CREATE_BULK_EVENTS()

This API creates multiple events for multiple transactions.

▲ Caution: Don't use this API for existing transactions that already have events associated with them. For performance reasons, bulk event APIs don't perform checks as to whether events for the transaction already exist. Therefore, use this API only to create events for new transactions that don't have any prior events created.

Information required for each event is inserted into the XLA_EVENTS_INT_GT table as described below, before the API is called.

The API checks all source information for valid application, legal entity, event entity, event number, and source IDs. It ensures that the required parameters are not null and also validates the event type and event status.

No validations are performed against the reference columns and event number.

Column Name	Data Type	Size	Required
ENTITY_CODE	VARCHAR2	30	Yes
APPLICATION_ID	NUMBER	15	Yes
LEDGER_ID	NUMBER	15	Yes
EVENT_ STATUS_CODE	VARCHAR2	30	Yes
EVENT_ TYPE_CODE	VARCHAR2	30	Yes
EVENT_DATE	DATE	-	Yes
TRANSACTION_ DATE	DATE	-	No



Column Name	Data Type	Size	Required
VALUATION_ METHOD	VARCHAR2	30	No
TRANSACTION_ NUMBER	VARCHAR2	240	No
BUDGETARY_ CONTROL_FLAG	VARCHAR2	1	No
SOURCE_ID_INT_1	NUMBER	15	No
SOURCE_ID_INT_2	NUMBER	15	No
SOURCE_ID_INT_3	NUMBER	15	No
SOURCE_ID_INT_4	NUMBER	15	No
SOURCE_ID_CHAR_1	VARCHAR2	30	No
SOURCE_ID_CHAR_2	VARCHAR2	30	No
SOURCE_ID_CHAR_3	VARCHAR2	30	No
SOURCE_ID_CHAR_4	VARCHAR2	30	No
SECURITY_ID_INT_1	NUMBER	15	No
SECURITY_ID_INT_2	NUMBER	15	No
SECURITY_ID_INT_3	NUMBER	15	No
SECURITY_ID_CHAR_1	VARCHAR2	30	No
SECURITY_ID_CHAR_2	VARCHAR2	30	No
SECURITY_ID_CHAR_3	VARCHAR2	30	No
REFERENCE_ NUM_ 1/ REFERENCE_ NUM _4	NUMBER	15	No
REFERENCE_ CHAR_ 1/ REFERENCE_ CHAR_4	VARCHAR2	30	No
REFERENCE_ DATE_ 1/ REFERENCE_ DATE_4	DATE	-	No

Procedure XLA_EVENTS_PUB_PKG.create_bulk_events



```
(p_source_application_id IN NUMBER DEFAULT NULL
,p_application_id IN NUMBER
,p_legal_entity_id IN NUMBER DEFAULT NULL
,p_ledger_id IN NUMBER
,p_entity_type_code IN VARCHAR2);
```

PL/SQL Requirements

When calling the Event API from a PL/SQL session, you must first initialize the session.

1.

Run the below query and note the role_name, role_guid, user_name, and user_guid.

```
select distinct
fff.function_name "function_name",
fsr.role_name "role_name",
fsr.role_guid "role_guid",
fs.user name "user name",
fs.user_guid "user_guid"
from
fnd objects fo,
fnd form functions fff,
fnd grants fg,
fnd session roles fsr,
fnd_sessions fs
where
fo.obj_name = 'XLA_TRANSACTION_ENTITIES' and
fo.object id = fff.object id and
fff.object_id = fg.object_id and
fg.role_name = fsr.role_name and
fsr.session_id = fs.session_id and
fs.user_name = &userName and
fg.role_name = &RoleName
when prompted with userName input: sample data is 'AP_SUPV_OPERATIONS'; when prompted with RoleName, sample
input is 'AP ACCOUNTS PAYABLE SUPERVISOR VISION OPERATIONS DATA';
```

Sample PL/SQL procedure to initialize the session. Replace **role_name**, **role_guid**, **user_name**, **user_guid** with results from the previous query.

```
SET SERVEROUTPUT ON SIZE 100000;

DECLARE

1_roleguids FND_TABLE_OF_VARCHAR2_4000 := FND_TABLE_OF_VARCHAR2_4000();
```



```
l_rolenames FND_TABLE_OF_VARCHAR2_4000 := FND_TABLE_OF_VARCHAR2_4000();

BEGIN
l_roleguids.extend(1);
l_rolenames.extend(1);
l_roleguids(1) := &role_guid;
l_rolenames(1) := &role_name;

if NVL(fnd_global.session_id,'-1') = '-1' then

fnd_global.initialize_session('X', 'X', l_roleguids, l_rolenames);

end if;

end;

When prompted to enter role_guid: sample input is like 'CB124ED92D967C2233CFD589C0AB110F'
when prompted to enter role name: sample input is 'AP ACCOUNTS PAYABLE MANAGER VISION OPERATIONS DATA
```

Updating Event APIs: Examples

The <code>update_Event_status()</code> API updates the event statuses of more than one event of a transaction. There is a set of overloaded APIs that can be used to update more than one attribute for an event. These APIs update accounting event type, event date, event status, event number, and reference information for an event. All of these API's use the <code>update Event()</code> API.

Note: Though these update event APIs retain the same name update_Event(), they use the PL/SQL feature of overloading to create unique procedures. In overloading, the input parameter names and types are distinct, resulting in unique procedures. Different columns are updated in the tables, depending on which procedure is called.

An event can be updated as long as it is not processed. Once an event is processed, you cannot update the event or the data associated with it.

This topic provides examples of the APIs that update events.

1. XLA_EVENTS_PUB_PKG.UPDATE_EVENT_STATUS()

In this example, the API updates the event status of one or more events within a transaction matching the specified criteria.

This API checks all source information for valid application, legal entity, event process category, and source IDs. It ensures that the required parameters are not null. The API also validates event status and if passed, event class and event type. The event date is truncated.

```
PROCEDURE XLA_EVENTS_PUB_PKG.update_event_status

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_class_code IN VARCHAR2 DEFAULT NULL
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security);
```

2. XLA_EVENTS_PUB_PKG.UPDATE_EVENT()

In this example the API updates multiple attributes of a single event. Using this API, the calling program can update event type, event date, and event status. An error code is returned if the update fails.



This API checks all source information for valid application, legal entity, event entity, and source IDs. The API ensures that the event ID is not null, that it belongs to the same transaction as the other transaction information being passed, and that the event hasn't already been accounted.

The parameters event type, event date, and event status are also validated if passed Not Null.

```
PROCEDURE XLA_EVENTS_PUB_PKG.update_event

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_event_number IN NUMBER
,p_reference_info IN
xla_events_pub_pkg.t_event_reference_info
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security);
,p_transaction_date_IN_DATE_DEFAULT_NULL)
```

XLA_EVENTS_PUB_PKG.UPDATE_EVENT()

In this example the API updates multiple attributes of a single event. Using this API, the calling program can update event type, event date, event status, and event number. An error code is returned if the update fails.

This API checks all source information for valid application, legal entity, event entity, and source IDs. No validations are performed against the event number but if no event number is passed, the routine populates the next highest event number for that transaction. The event date is truncated.

The API ensures that the event ID is not null, that it belongs to the same transaction as the other transaction information being passed, and that the event hasn't already been accounted. The parameters event type, event date, and event status are also validated if passed Not Null.

```
PROCEDURE XLA_EVENTS_PUB_PKG.UPDATE_EVENT

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_event_number IN NUMBER
,p_valuation_method IN VARCHAR2
,p_transaction_date IN DATE DEFAULT NULL
,p_security_context IN xla_events_pub_pkg.t_security);
```

4. XLA_EVENTS_PUB_PKG.UPDATE_EVENT()

In this example the API updates multiple attributes of a single event. Using this API, the calling program can update event type, event date, event status, and the event's reference information. An error code is returned if the update fails.

This API checks all source information for valid application, legal entity, event entity, and source IDs. No validations are performed on the reference information.

The API ensures that the event ID is not null, that it belongs to the same transaction as the other transaction information being passed, and that the event hasn't already been accounted. The parameters event type, event date, and event status are also validated if passed Not Null.

```
(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_reference_info IN xla_events_pub_pkg.t_event_reference_info
```



```
,p_valuation_method IN VARCHAR2
,p_transaction_date IN DATE DEFAULT NULL
,p_security_context IN xla_events_pub_pkg.t_security);
```

XLA_EVENTS_PUB_PKG.UPDATE_EVENT()

In this example the API updates multiple attributes of a single event. Using this API, the calling program can update event type, event date, event status, event number, and the event's reference information. An error code is returned if the update fails.

Note: This API updates both the event's event number and reference information.

This API checks all source information for valid application, legal entity, event entity, and source IDs. The API ensures that the event ID is not null, that it belongs to the same transaction as the other transaction information being passed, and that the event hasn't already been accounted.

The parameters event type, event date, and event status are also validated if passed Not Null. No validations are performed against the event number and reference information, but if no event number is passed, the routine populates the next highest event number for that transaction.

```
PROCEDURE XLA_EVENTS_PUB_PKG.update_event

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_event_status_code IN VARCHAR2 DEFAULT NULL
,p_valuation_method IN VARCHAR2
,p_transaction_date IN DATE DEFAULT NULL)
,p_security_context IN xla_events_pub_pkg.t_security);
```

6. XLA EVENTS PUB PKG.UPDATE BULK EVENT STATUSES

This API updates the event status of multiple events. Before calling this API, users must populate the XLA_EVENTS_INT_GT table with the following:

- application_id
- entity_code
- ledger_id
- event_id
- event_status_code

This API updates the events in the XLA_EVENTS table to the new status.

This API validates the application ID, event entity, event ID, and event status. The status of both the new and old status cannot be Processed. The new status must be a valid event status.

```
Procedure XLA_EVENTS_PUB_PKG.update_bulk_event_statuses
(p_application_id IN INTEGER);
```



Deleting Event APIs: Examples

This topic provides examples of the APIs that delete events.

1. XLA_EVENTS_PUB_PKG.DELETE_EVENT()

This API deletes an unaccounted event based on its event identifier. The API returns an error code if the delete fails.

The API checks all source information for valid application, legal entity, event process category, and source IDs. This ensures that the required parameters are not null, the event ID belongs to the same transaction as other transaction information being passed, and that the event hasn't been accounted.

```
PROCEDURE XLA_EVENTS_PUB_PKG.delete_event

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_id IN NUMBER
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security);
```

2. XLA_EVENTS_PUB_PKG.DELETE_EVENTS()

This API deletes all events for a transaction that meet the specified criteria. When called, events that belong to the given accounting event class, event type, and event date are deleted.

The API checks all source information for valid application, legal entity, event process category, and source IDs. It ensures that the required parameters are not null and if passed, validates the event type and event status.

The function returns the number of events deleted.

```
FUNCTION XLA_EVENTS_PUB_PKG.delete_events

(p_event_source_info IN xla_events_pub_pkg.t_event_source_info
,p_event_class_code IN VARCHAR2 DEFAULT NULL
,p_event_type_code IN VARCHAR2 DEFAULT NULL
,p_event_date IN DATE DEFAULT NULL
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)
RETURN INTEGER;
```

3. XLA_EVENTS_PUB_PKG.DELETE_ENTITY()

This API deletes a row from the XLA_TRANSACTION_ENTITIES table. The routine checks if there are still events associated with the transaction. If yes, the routine does nothing and returns 1; otherwise, it deletes the transaction in the XLA_TRANSACTION_ENTITIES table and returns 0.

No validations exist for this API.

If a transaction is deleted, users must call the DELETE_ENTITY API to delete the row in the XLA_TRANSACTION_ENTITIES table.

```
Function XLA_EVENTS_PUB_PKG.delete_entity
(p_source_info IN xla_events_pub_pkg.t_event_source_info
,p_valuation_method IN VARCHAR2
,p_security_context IN xla_events_pub_pkg.t_security)
RETURN INTEGER;
```



4. XLA_EVENTS_PUB_PKG.DELETE_BULK_EVENTS()

This API deletes multiple events. Before calling this API, users must populate the XLA_EVENTS_INT_GT table with the following:

- application_id
- entity_code
- ledger_id
- event_id
- event_status_code

The API deletes events from the XLA_EVENTS table.

This API validates the application ID, event process category, and event ID. The status of the event to be deleted cannot be processed.

Procedure XLA_EVENTS_PUB_PKG.delete_bulk_events
(p_application_id IN INTEGER);

Common Parameters: Explained

This topic provides details on the parameters common to many event APIs.

This topic includes information on the following types of parameters:

- Transaction identifiers
- Contextual information
- Transaction security identifiers
- Transaction reference information
- Event information

Transaction Identifiers

The table below describes transaction identifier attributes.

Parameter Name	Туре	Description
p_ transaction_ number	Varchar2(240)	Transaction number of the event-based transaction. Acts as the user transaction identifier and serves as a reference for the transaction.
p_ event_ source_info	xla_ events_ pub_ pkg.t_event_ source_info	System transaction identifiers

Contextual Information

The table below describes contextual information details.



Туре	Description
Integer	Internal identifier of the application that generates the document or transaction. This may be different from the application that generates and owns the accounting for the corresponding event. If no value is provided, the default is p_
	application_ id. Used for Oracle subledgers. This parameter is unlikely to be used for non-Oracle systems.
Number	Application internal identifier.
Number	Legal entity internal identifier.
Number	Ledger internal identifier.
Varchar2(30)	Valuation method used for securing a transaction. Some applications secure their transactions by valuation method. Used for Oracle subledgers. This parameter is unlikely to be used for non-Oracle systems.
	Number Number Number

Transaction Security Identifiers

The table below describes transaction security identifier attributes.

Parameter Name	Туре	Description
p_ security_ context	xla_ events_ pub_ pkg.t_ Security	Security context information for the transaction that has created the events.

Transaction Reference Information

The table below describes transaction reference parameter attributes.

Parameter Name	Туре	Description
p_ array_ reference_ info	xla_ events_ pub_ pkg.t_array_ event_ reference_ info	Array of optional reference information for multiple events. These are stored with the events.
p_ reference_ info	xla_ events_ pub_ pkg.t_event_ reference_ info	Optional reference information for a particular event.



Event Information

The table below describes event information parameter attributes.

Parameter Name	Туре	Description
p_ entity_ type_code	Varchar2(30)	Entity type internal code.
p_ event_ class_code	Varchar2(30)	Event class internal code.
p_ event_ type_code	Varchar2(30)	Event type internal code
p_event_id	Number	Event internal identifier.
p_event_date	Date	Event accounting date.
p_ event_ status_code	Varchar2(1)	External status code for an event.
p_event_number	Number	Event sequence number within a transaction. Events are ordered by this sequence number for accounting.
p_ array_ entity_ source_info	xla_ events_ pub_ pkg.t_array_ entity_ source_info	Array of transaction source ID information as stamped on the entity.
p_ array_ event_ type_code	xla_ events_ pub_ pkg.t_array_ event_ type_code	Array of internal codes for the event type as defined by applications.
p_ array_ event_date	xla_ events_ pub_ pkg.t_array_ event_date	Array of accounting dates for events.
p_ array_ event_ status_code	xla_ events_ pub_ pkg.t_array_ event_ status_code	Array of external status codes for events.
p_ array_ event_number	xla_ events_ pub_ pkg.t_array_ event_number	Array of event sequence numbers within a transaction. Events are ordered by these sequence numbers for accounting.
p_ array_ entity_ event_info	xla_ events_ pub_ pkg.t_array_ entity_ event_info	Array of combined entity and event attributes.

PL/SQL Data Types: Explained

The following are the predefined PL/SQL data structures available in the XLA_EVENTS_PUB_PKG package.

PL/SQL Record Structures

Transaction Source Information



```
TYPE t_event_source_info IS RECORD
(source_application_id NUMBER DEFAULT NULL
,application_id NUMBER
,legal_entity_id NUMBER
,ledger_id NUMBER
,entity_type_code VARCHAR2(30)
,transaction_number VARCHAR2(240)
,source_id_int_1 NUMBER
,source_id_int_2 NUMBER
,source_id_int_2 NUMBER
,source_id_int_3 NUMBER
,source_id_int_4 NUMBER
,source_id_int_4 NUMBER
,source_id_char_1 VARCHAR2(30)
,source_id_char_2 VARCHAR2(30)
,source_id_char_3 VARCHAR2(30)
,source_id_char_4 VARCHAR2(30));
```

The table below provides descriptions on select attributes listed above.

Attribute Table #1

Attribute	Description
source_ application_ id	Internal identifier of the application that generates the document or transaction. This may be different from the application that generates and or owns the accounting for the corresponding event. Source applications don't have be registered as subledger applications.
	If no value is provided, the default is application_id.
	Used for Oracle subledgers. This parameter is unlikely to be used for non-Oracle systems.
application_id	Application transaction owner identifier.
legal_ entity_id	Transaction legal entity identifier.
ledger_id	Transaction ledger identifier.
entity_ type_code	Entity code as defined by applications during set up.
transaction_ number	Transaction number of the transaction that has created the events. The transaction number serves as a reference for the transaction.
source_ id_xxx_n	Generic columns that store the identifier for the transaction in the transaction table.

Event Reference Information

```
TYPE t_event_reference_info IS RECORD
(reference_num_1 NUMBER
,reference_num_2 NUMBER
,reference_num_3 NUMBER
,reference_num_4 NUMBER
,reference_char_1 VARCHAR2(240)
,reference_char_2 VARCHAR2(240)
,reference_char_3 VARCHAR2(240)
,reference_char_4 VARCHAR2(240)
,reference_date_1 DATE
,reference_date_2 DATE
```



```
,reference_date_3 DATE
,reference_date_4 DATE);
```

Note: See Attribute Table #2 for descriptions on select attributes.

Event Information

```
TYPE t_event_info IS RECORD
(event_id NUMBER
,event_number NUMBER
,event_type_code VARCHAR2(30)
,event_date DATE
,event_status_code VARCHAR2(1)
,process_status_code VARCHAR2(1)
,reference num 1 NUMBER
,reference_num_2 NUMBER
,reference_num_3 NUMBER
,reference num 4 NUMBER
,reference char 1 VARCHAR2(240)
,reference_char_2 VARCHAR2(240)
,reference_char_3 VARCHAR2(240)
,reference_char_4 VARCHAR2(240)
,reference_date_1 DATE
,reference_date_2 DATE
,reference_date_3 DATE
,reference_date_4 DATE);
```

Note: See Attribute Table #2 for descriptions on select attributes.

The table below provides descriptions on select attributes listed above.

Attribute Table #2

Attribute	Description
event_type_code	Code for the event type of the event, as defined during set up.
event_date	Event accounting date.
event_id	Event internal identifier.
event_number	Event sequence number for the event within a transaction. Events are processed in the order of their event number.
event_ status_code	Status code for the event. The event's external status and is used by implementors.
transaction_ number	Transaction number of the transaction that has created the events. The transaction number serves as a reference for the transaction.
source_id_xxx_n	Generic columns that store the identifier for the transaction in the transaction table.
reference_xxx_n	Generic columns that store reference information for the event.



Attribute	Description
valuation_ method	Valuation method code used as a security context for applications that support the valuation method.
	Used for Oracle subledgers. This parameter is unlikely to be used for non-Oracle systems.
security_ id_xxx_n	Security contexts.

Security Context Information

Security context information is optional and restricts a user's access to several features in the Oracle Fusion Accounting Hub. In conjunction with the Fusion Security model, the security context information determines:

- Which events the Create Accounting process processes.
- Which events and entries are visible to the user on the screens and reports.

If used, the securing organization's IDs should be populated in the attributes. As an example, a securing organization can be a business unit or a ledger.

Use the following record structure to pass security context information through event APIs. This structure is defined in XLA_EVENTS_PUB_PKG package.

```
TYPE t_security IS RECORD
(security_id_int_1 NUMBER
,security_id_int_2 NUMBER
,security_id_int_3 NUMBER
,security_id_int_3 NUMBER
,security_id_char_1 VARCHAR2(30)
,security_id_char_2 VARCHAR2(30)
,security_id_char_3 VARCHAR2(30));
```

The table below provides descriptions for the attributes listed above.

Attribute Table #3

Attribute	Description
security_id_int_n	Security context information in INTEGER type.
security_ id_char_n	Security context information in VARCHAR type.

PL/SQL Table Structure

Array of Information based on above Structures

```
TYPE t_array_event_reference_info IS TABLE OF t_event_reference_info
TYPE t_array_event_info IS TABLE OF t_event_info
TYPE t_array_event_source_info IS TABLE OF t_event_source_info
```

Other Array Structures

```
TYPE t_array_event_type IS TABLE OF VARCHAR2(30)

TYPE t_array_event_date IS TABLE OF DATE

TYPE t_array_event_status_code IS TABLE OF VARCHAR2(1)
```



```
TYPE t_array_entity_id IS TABLE OF NUMBER
TYPE t_array_event_id IS TABLE OF NUMBER
TYPE t_array_event_number IS TABLE OF NUMBER
```

Constants: Explained

The Oracle Fusion Accounting Hub uses predefined accounting event status Constants. These constants are defined in the XLA EVENTS PUB PKG package.

Event Statuses and Constants

The table below lists event statuses and the corresponding constants that must be used when employing event APIs.

Event Status	Constant
Incomplete	XLA_ EVENTS_ PUB_ PKG.C_EVENT_ INCOMPLETE
Unprocessed	XLA_ EVENTS_ PUB_ PKG.C_EVENT_ UNPROCESSED
No Action	XLA_ EVENTS_ PUB_ PKG.C_ EVENT_ NOACTION
Processed	XLA_ EVENTS_ PUB_ PKG.C_ EVENT_ PROCESSED

Create Accounting Integration

Create Accounting Integration: Overview

The Oracle Fusion Accounting Hub enables customers to extend and customize the integration with the Create Accounting process by invoking API calls. The Accounting Hub code structure enables the integration of customized business logic at various stages during the accounting process.

Specifically, you must customize the xla_acct_hooks_pkg if you want to add custom logic.

You can add logic to these APIs to instruct the Create Accounting process to perform at the indicated step.

- Preaccounting
- Create Accounting extract
- Postprocessing
- Postaccounting

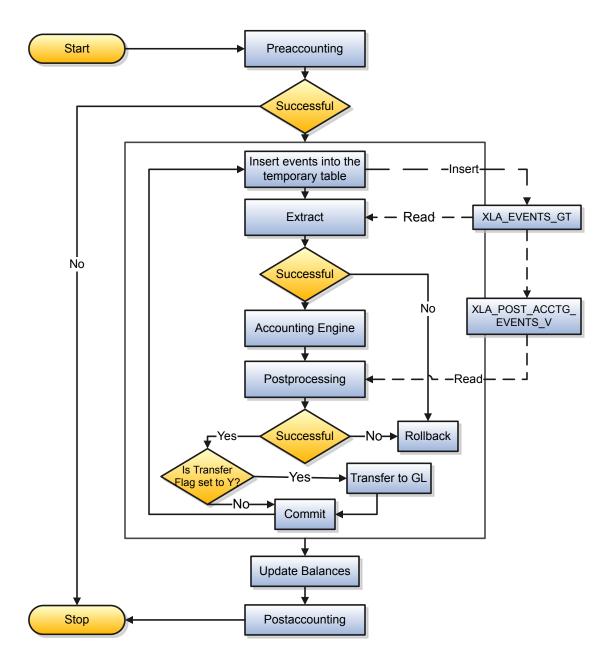


Timing and Positioning of APIs: Explained

This section describes the life cycle of the Create Accounting process.

Create Accounting Process Life Cycle

The figure below shows the life cycle of the Create Accounting process.





Preaccounting

The preaccounting logic is called at the beginning of the Create Accounting process before selecting accounting events for processing.

Example: This can be used to evaluate a condition to identify transactions for processing.

Extract

The extract logic is called for all modes of accounting, final and draft, in batch as well as document mode. Called after selecting the next set of events to be processed, but before any processing is done. Events selected for processing are made available to custom logic through the global temporary table XLA_EVENTS_GT. Implementors adding code to this API can use it to populate transaction objects based on information in the XLA_EVENTS_GT table.

Example: Use this to populate transaction objects.

Postprocessing

The postprocessing logic is called after creating subledger journal entries for each processing unit (commit unit). Custom logic within this API can use the view XLA_POST_ACCTG_EVENTS_V to determine which accounting events were successfully accounted by the Create Accounting process.

Example: Use this to update posted flags on the transaction distributions to indicate whether the transaction was accounted successfully.

Postaccounting

The postaccounting logic is called after subledger journal entries are successfully committed in the database.

Examples: Use this to send a notification to a system administrator that the accounting process has completed.

Or, use this to export the supporting reference balances to a third-party reconciliation tool for comparison to source system totals to ensure accounting is complete.

XLA_POST_ACCTG_EVENTS_V

The table below describes the view structure.

Column Name	Null?	Data Type	Description
APPLICATION_ID	Not Null	NUMBER(18)	Internal identifier for the application to which the event belongs
ENTITY_CODE	Not Null	VARCHAR2(30)	Event entity type code
ENTITY_ID	Not Null	NUMBER(18)	Internal identifier for the entity representing the actual document
EVENT_ CLASS_CODE	Not Null	VARCHAR2(30)	Event class code
EVENT_DATE	Not Null	DATE	Event or accounting date for the event



Column Name	Null?	Data Type	Description
EVENT_ID	Not Null	NUMBER(18)	Event internal identifier
EVENT_NUMBER	Not Null	NUMBER(18)	Event number assigned to the event within the document
EVENT_ STATUS_CODE	Not Null	VARCHAR2(1)	Event status code
EVENT_TYPE_CODE	Null	VARCHAR2(30)	Code for the event type that classifies the event being created
LEDGER_ID	Not Null	NUMBER(18)	Ledger internal identifier to which the event belongs
LEGAL_ ENTITY_ID	Null	NUMBER(18)	Internal identifier for the legal entity
PROCESS_ STATUS_ CODE	Not Null	VARCHAR2(1)	Event process code
REFERENCE_NUM_1	Null	NUMBER	Reference information
REFERENCE_ NUM_2	Null	NUMBER	Reference information
REFERENCE_ NUM_3	Null	NUMBER	Reference information
REFERENCE_ NUM_4	Null	NUMBER	Reference information
REFERENCE_ CHAR_1	Null	VARCHAR2(240)	Reference information
REFERENCE_ CHAR_2	Null	VARCHAR2(240)	Reference information
REFERENCE_ CHAR_3	Null	VARCHAR2(240)	Reference information
REFERENCE_ CHAR_4	Null	VARCHAR2(240)	Reference information
REFERENCE_ DATE_1	Null	DATE	Reference information
REFERENCE_ DATE_2	Null	DATE	Reference information
REFERENCE_ DATE_3	Null	DATE	Reference information
REFERENCE_ DATE_4	Null	DATE	Reference information
SOURCE_ID_INT_1	Null	NUMBER(18)	Placeholder column that stores internal identifier of the document being represented by the entity



Column Name	Null?	Data Type	Description
SOURCE_ ID_INT_2	Null	NUMBER(18)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ ID_INT_3	Null	NUMBER(18)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ID_INT_4	Null	NUMBER(18)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ID_CHAR_1	Null	VARCHAR2(30)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ ID_CHAR_2	Null	VARCHAR2(30)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ ID_CHAR_3	Null	VARCHAR2(30)	Placeholder column that stores internal identifier of the document being represented by the entity
SOURCE_ID_CHAR_4	Null	VARCHAR2(30)	Placeholder column that stores internal identifier of the document being represented by the entity
TRANSACTION_ NUMBER	Null	VARCHAR2(240)	Transaction number given to the document by the products owning the document

Parameter Specifications: Explained

This section describes parameters for the Create Accounting process integration points.

Preaccounting Parameters

The table below describes the parameters for xla_acct_hooks_pkg.preaccounting().



Parameter Name	Description
P_ APPLICATION_ ID	Application identifier for which the Create Accounting process is submitted. The custom logic checks this parameter first to see if it's the wanted application.
P_LEDGER_ID	Ledger identifier for which the Create Accounting process is submitted.
P_ PROCESS_ CATEGORY	Process category specified by you when launching the Create Accounting process.
P_END_DATE	End date specified by you.
P_ ACCOUNTING_ MODE	Indicates the mode in which the Create Accounting process is submitted.
P_ REPORT_ REQUEST_ID	Request ID of the Create Accounting process submitted.
P_ VALUATION_ METHOD	Valuation method specified by you.
	Relevant for internal Oracle applications.
P_SECURITY_ID_INT_1	Security context values as passed in as a parameter for the Create Accounting process.
P_SECURITY_ID_INT_2	
P_SECURITY_ID_INT_3	
P_SECURITY_ID_CHAR_1	
P_SECURITY_ID_CHAR_2	
P_ SECURITY_ ID_CHAR_3	

Extract Parameters

The table below describes the parameters for xla_acct_hooks_pkg.extract().

Parameter Name	Description
P_ APPLICATION_ ID	Application identifier for which the Create Accounting process is submitted. The custom logic checks this parameter first to see if it's the wanted application.
P_ ACCOUNTING_ MODE	Indicates the mode in which the Create Accounting process is submitted

Postprocessing Parameters

The table that describes the parameters for xla_acct_hooks_pkg.postprocessing() is the same as the table that defines the parameters for extract.



Postaccounting Parameters

The table that describes the parameters for xla_acct_hooks_pkg.postaccounting() is the same as the table that defines the parameters for preaccounting.

FAQs for Create Account Integration

How can I integrate non-Oracle applications or customizations with the Oracle Fusion Accounting Hub?

To integrate non-Oracle applications or customizations with the Oracle Accounting Hub, you can add logic to the preaccounting, extract, postprocessing, or postaccounting APIs in the xla_acct_hook_pkg.

To improve the performance, Oracle recommends that you check the Application ID as the first step in your logic. Do not continue if it does not match.

What happens if there is an exception to the API logic?

If the API logic raises an exception, then that exception will abort the Create Accounting process.

If the logic that raises the exception is in preaccounting, then the Create Accounting process will immediately rollback and exit without doing any accounting.

If the logic that raises the exception is in extract or postprocessing, then any accounting done for the current set of transactions will be aborted and rolled back, and the worker will exit. However, other workers may continue processing.

If the logic that raises the exception is in postaccounting, then the Create Accounting process will rollback and exit. However, this will only rollback anything done by the postaccounting logic. It will not rollback any accounting done by the Create Accounting process.

Can I receive a notification of the status of a Create Accounting process?

In Oracle Fusion Accounting Hub you can add logic to the postaccounting API to notify you of the status of the Create Accounting process.

Manage Accounting Rules



Creating Accounting Method: Explained

Accounting methods group subledger journal entry rule sets to define a consistent accounting treatment for each accounting event class and accounting event type for all subledger applications. This grouping allows a set of subledger journal entry rule sets to be assigned collectively to a ledger.

For example:

- A subledger accounting method entitled US GAAP can be defined to group subledger journal entry rule sets that adhere to and comply with US Generally Accepted Accounting Principles (GAAP) criteria.
- By assigning a different subledger accounting method to each related ledger, you can create multiple accounting representations of transactions.

Accounting rules can be defined with either a top down, or a bottom up approach.

- Top Down: Define the accounting method, followed by components of each rule that must be assigned to it.
- Bottom Up: Define components for each rule and then assign them as required.

The Create Accounting process uses the accounting method definition with active journal entry rule set assignments to create subledger journal entries.

When an accounting method is initially defined its status changes to Incomplete. The status will also be Incomplete after modifying a component of any accounting rule associated to the assigned journal entry rule set.

▲ Caution: The accounting method must be completed, by activating its journal entry rule set assignments, so that it can be used to create accounting.

The following definitions are used to define the journal entries, and are applied as updates to the accounting method:

- Updates to the predefined accounting method
- Assignment of journal entry rule sets for an accounting event class and accounting event type from the accounting methods page
- Assignment of accounting methods to ledgers
- Activation of subledger journal entry rule set assignments

Updates on Predefined Accounting Method

You may update a predefined accounting method by end dating the existing assignment and creating an assignment with an effective start date.

Assignment of Journal Entry Rule Set for Accounting Event Class and Accounting Event Type

You create the assignment of a journal entry rule set for an accounting event class and accounting event type using the accounting method page.

The following should be considered for assigning rule sets:

- If the accounting method has an assigned chart of accounts you can select journal entry rule sets that:
 - Use the same chart of accounts
 - Are not associated with any chart of accounts



Select an option to assign existing journal entry rule sets or create one.

Assignment of Accounting Methods to Ledgers

If the accounting method has an assigned chart of accounts, it may only be used by ledgers that use the same chart of accounts.

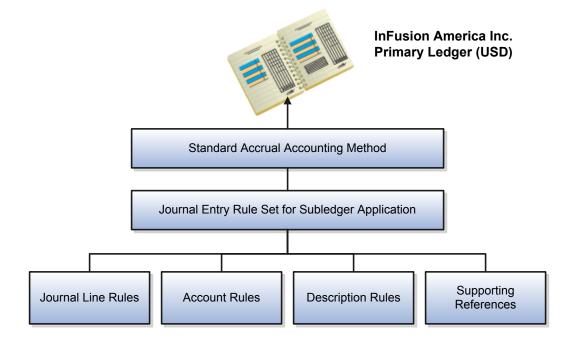
If the accounting method does not have an assigned chart of accounts, the accounting method can be assigned to any ledger.

Activation of Subledger Journal Entry Rule Set Assignments

You can activate the subledger journal entry rule set assignments from the Accounting Method page. You can also submit the Activate Subledger Journal Entry Rule Set Assignments process to validate and activate your accounting set ups.

Fusion Setup Flow

The figure below shows the relationship of components making up an accounting method as described in the above text.



Creating Subledger Journal Entry Rule Sets: Explained

Subledger journal entry rule sets provide the definition for generating a complete journal entry for an accounting event. Select the option to define the subledger journal entry rule set for a particular accounting event class or accounting event type.

If you are using multiple ledgers to meet divergent and mutually exclusive accounting requirements, you can vary journal entry rule sets by ledger. Each of the subledger journal entry rule sets can meet a specific type of accounting requirements.



For example, use US Generally Accepted Accounting Principles (GAAP) oriented subledger journal entry rule sets for a ledger dedicated to US GAAP reporting. Use French statutory accounting conventions for a ledger dedicated to French statutory reporting. These two sets of definitions have differences based on the setup of the various components that make up their subledger journal entry rule sets.

Predefined subledger journal entry rule sets are provided for all Oracle subledgers. If specific requirements are not met by predefined subledger journal entry rule sets, create a copy of the predefined definitions, rename, and modify the copied definitions and their assignments.

Subledger journal entry rule set assignments can be made at two levels, header and line. The following are the subcomponents of a subledger journal entry rule set:

- Description rules
- Journal line rules
- Account rules

Assignment at Header Level

Header assignments define subledger journal header information and line assignments define journal line accounting treatment.

A header assignment includes the following:

- Accounting date (required)
- Accrual reversal accounting date (optional)
- Description rule (optional)

Assignment at Line Level

You can define multiple subledger journal entry rule sets for an accounting event class or accounting event type. Using the line assignment of the journal entry rule set assigned to the accounting event class or type, a single journal entry is generated per accounting event per ledger.

The following can be assigned to a journal entry line:

- Journal line description rule
- Journal line rule
- Account rule
- Supporting references

Assignment of Description Rules

If a description rule is defined with sources, the sources must also be assigned to the accounting event class that is assigned to the journal entry rule set. The description rule may be assigned at either the header or line level of the journal entry or to both levels.

Assignment of Journal Line Rules

When assigning the journal line rule, you must identify the line type: Gain, Loss, Gain or Loss, Credit, or Debit. The journal line rule must be assigned to the same accounting event class as the one assigned to the subledger journal entry rule set.



When assigning a journal line rule that is enabled for accounting for a business flow, the account combination and certain accounting attribute values are copied from its related journal line having the same business flow class as the current line. Optionally, copy the description rule into the current line instead of assigning a separate description rule.

When assigning a journal line rule that is enabled to copy from the corresponding line within the same journal entry, you have the option to copy the account combination, the segment value, or the line description from the corresponding line into the current line.

Assignment of Account Rules

The account rule assignment defines which accounts are used for the subledger journal line. If the account rule is set up with a chart of accounts, it must have the same chart of accounts as the one assigned to the journal entry rule set. When account rules are defined with sources, the sources must also be assigned to the accounting event class that is assigned the journal entry rule set.

There are two types of account rules:

- Account Combination Rule: Assign an account combination rule to derive the account combination.
- Segment Rule: Assign a segment rule to derive a specific segment of an account. For example, a cost center or a
 natural account segment.

Assignment of Supporting References

Supporting references may be used to capture transaction values on journal entry lines. A supporting reference can be used on a journal entry rule set only if it's assigned a source from the event class of the journal entry rule set.

Journal Line Rules: Explained

Journal line rules are defined within the context of accounting event classes. A journal line rule can be used in a subledger journal entry rule set that has the same event class. You may also assign conditions to the journal line rule.

Journal Line Rules

Journal line rules are assigned to journal entry rule sets.

To create a journal line rule, select values for options such as:

Side (Debit, Credit, Gain or Loss)

For example, when an Oracle Fusion Payables invoice is generated, the liability account should normally be credited. The journal line rule must therefore specify the Side option as Credit. On the other hand, the payment of the Payables invoice must be accounted with a debit to the liability account. A separate journal line rule must be defined to create this debit line.

- Merge Matching Lines: To summarize subledger journal entry lines within each subledger entry. Journal entry lines with matching criteria are merged.
- Accounting Class
 - Select an accounting class to classify journal entry lines.



- For example, when a validated Payables invoice is accounted, the Item Expense and Liability journal lines are created. In this case, the journal line rules used in the accounting rules are assigned Item Expense and Liability accounting classes respectively.
- Conditions: To restrict the use of a journal line rule by controlling when a particular journal line rule is used by the Create Accounting process.
- Accounting Attributes: When creating a journal line rule, accounting attribute assignments are automatically
 established based on the default accounting attribute assignments for that journal line rule's accounting event class.
 You can override this default mapping of standard sources to accounting attributes. The list of values for the source
 override includes all sources assigned to the accounting attribute for the event class associated with the journal line
 rule.
- Advanced Options
 - The Subledger Gain or Less Option: Applies only to amount calculations for the primary ledger. Gain or loss amounts are not converted to reporting currency or nonvaluation method secondary ledgers. If the option is selected, the journal line holds the gain or loss amounts calculated by the subledger.
 - The gain or loss amount is calculated as the difference in applied amounts due to fluctuations in conversion rates, based upon conversion to the ledger currency. Foreign exchange gain or loss amounts occur when two related transactions, such as an invoice and its payment, are entered in a currency other than the ledger currency, and the conversion rate fluctuates between the times that the two are accounted.
 - The Rounding Class Option: Along with transaction rounding, groups journal lines together and calculates transaction rounding. Subledger transaction rounding differences can occur when a transaction has multiplerelated applied-to transactions, such as a Receivables invoice that has multiple associated receipts.
 - The Link Journal Lines Option: Determines whether the journal line rule is set up to establish a link between the accounting of transactions that are related both within the same application, and across applications. The alternatives are described in this table:

Link Journal Lines Option	Description
None	No link is established.
Copy from corresponding line	Build account for a journal line using segments from the offsetting entry of the current journal line. For example, when the business process requires that a cost center incurring an expense must also bear the invoice liability and cash outlay.
Business flow	Link logically related business transactions. For example, when recording the closing of a loan, you can link to the account that was used to book the loan origination. Journal line rules that are linked must also be assigned the same business flow class.

Defining Conditions for Journal Line Rules

You may set conditions to specify whether the journal line rule are used to create a subledger journal entry line. If the conditions are true, the line rule is used to create a subledger journal entry line. Use sources to create these conditions.



For example, you can set up a condition that creates a journal line to record tax, only if there is tax for an invoice. The line type and account class mentioned here are examples of sources.

- The condition for a Payables invoice tax journal line rule could be:
 - Where Line Type = Tax
 - When this condition is true, there is tax for a payables invoice line. A journal entry line is created to record the accounting impact of the tax.
- Similarly, the condition for an invoice tax journal line rule could be:
 - Where Account Class = Tax
 - In this case, if there is an account class of Tax, the journal line is used to record the accounting impact of the tax.

Another example is a condition that creates a journal line for freight when there are freight charges on an invoice.

Journal line rule conditions determine whether a journal line rule and its associated account rules and description rules are used to create the subledger journal entry line. If the conditions of all the journal line rules assigned to the journal entry rule set are not met, the transaction is processed without the creation of any subledger journal entries, and the event status is set to Processed.

Note: Constant values that are used in any Conditions region must not contain the following characters:

- •
- ,
- .
- (
-)
- ,

For example, in the condition "Project Type" = ABC (123), the constant value following the equal sign, ABC (123), contains restricted characters () that enclose 123 and is invalid.

Account Rules: Explained

Account rules are used to determine the accounts for subledger journal entry lines. In addition, you can specify the conditions under which these rules apply. Using these capabilities, you can develop complex rules for defining accounts under different circumstances to meet your specific requirements. You can define account rules for an account, segment, or value set.

Account Rules by Account

Define account rules by account to determine the entire account combination. For example, an account rule defined by account can be used to determine the complete supplier liability account in Oracle Fusion Payables.

Account Rules by Segment

Define segment rules to derive a specific segment of the general ledger account. For example, a particular segment like the company segment can be determined from the distribution account.



Another segment can be determined with the use of a constant value. Creating the account one segment at a time offers greater flexibility, but also requires more setup.

Use both segment based and account based rules to derive a single account. Segment-specific rules are used, where they are defined, and take the remaining values from an account-based rule. For example, you can select an account rule which is for all segments and also separately select a rule which is for one particular segment. Segment-specific rules take precedence over the all segments account based rule.

Combine account rules with segment rules. In this case, the segment value is derived from the segment rule to override the corresponding segment of the account. If the segment rule has conditions associated with the priorities and none are met, no override occurs and the segment value is derived from the account rule.

Note:

- If the returned account is end dated with a date that is the same or before the subledger journal entry
 accounting date, and an alternate account is defined in the general ledger, the alternate account is used. The
 original account is stored on the journal line for audit purposes
- If the alternate account is invalid, and the **Post Invalid Accounts to Suspense Account** option is selected in the Create Accounting process, then a suspense account is used. An error message is displayed if a valid suspense account is not available.

Account Rules by Value Sets

In the absence of a chart of accounts, you may define account rules based upon value sets. This enables you to share the same rule between more than one chart of accounts if the segments in these charts of accounts share the same value set.

Sharing Account Rules across Applications

You may share account rules across applications in the following ways.

- Assign an account rule from the same or a different application to a journal line rule in the subledger journal entry rule set. For example, to derive an expense account for journal line rule Expense, assign the Projects Cost Account rule owned to the Payables journal line rule Expense.
- Create an account rule based on an account rule from another application and assign it to a journal line rule. For example, you may create an account rule Invoice Expense Account referencing Project Cost Account assigned in the Priorities region. You may attach the Invoice Expense Account rule to the journal line rule Expense in the journal entry rule set.

✓ Note:

- To share an account rule across applications, all sources used by the account rule must be available for the event class.
- If the sources are available, an account rule is assigned to a journal line rule in the journal entry rule set.
 Verification occurs to confirm that all sources used by the account rule are available for the journal line rule accounting event class. Journal line rules are only available if the sources are shared; such as reference objects.

Account Rules and Mapping Sets

Mapping sets can be used to associate a specific output value for an account or segment. You can use mapping sets in account rules to build the account.



Account Rules Conditions

In the account rules you may specify conditions for each rule detail line. Priorities determine the order in which account rule conditions are examined. When the condition is met, the rule associated with that priority is used. Depending on which of the defined conditions is met, a different account rule detail is employed to create the account.

The Create Accounting process evaluates conditions based on the priority of the rule detail. When the condition is met, the rule detail is applied.

Creating Account Rules: Points to Consider

You can define an account rule using the following rule types:

- · Account combination
- Segment
- Value Set

Account Combination Rules

Set up account combination rules based upon the following value types:

- 1. Source Value Type: Derive the account combination by specifying a source.
 - Sources that have been set up as accounts can be assigned to an account combination rule. Subledger Accounting then obtains the account combination identifier from the source.
- 2. Constant Value Type: Establish the account as a constant value.
 - For example, the constant could be a completed account combination from the chart of accounts specified. An example is the account combination, 01.000.2210.0000.000. This is the simplest way to derive an account.
- 3. Mapping Set Value Type: Derive the account combination by referencing a mapping set.
 - Set up a mapping set to determine the complete account combination from the chart of accounts specified.
- 4. Account Rule Value Type: Derive the account by referencing another account rule.
 - The chart of accounts is optional when defining this type of rule. If the account rule has a chart of accounts assigned, then all the related account rules must use the same or no chart of accounts.
- Note: A chart of accounts must be specified for account combination rules using constants.

Segment Rules

Set up segment rules as follows:

- When a chart of accounts is specified, create a rule to derive the value for a specific segment from the chart of accounts.
- If the chart of accounts is not specified, create a rule to derive the value for an account segment with a specific qualifier.

Set up segment rules using the same methods discussed in the preceding Account Combination Rules section. By specifying different value types, users can select the way in which the segment value is derived.

Note: A chart of accounts must be specified for segment rules using constants.



Value Set Rules

Value set based rules can be created when a chart of accounts is not specified, enabling you to share the same rule between more than one chart of accounts. But, only if the segments in these charts of accounts share the same value set.

Set up value set based rules using the same methods discussed in the preceding Account Combination Rules section.

Mapping Sets: Explained

Mapping sets provide an efficient way to define a segment or account combination value for one or more transaction or reference attribute values. Using such input and output mappings is simpler than using complex conditions on account rules.

Based on the value of the source input, a single segment or a full account is derived.

Examples of source input:

- Transaction attributes
- Reference attributes

With mapping sets you can:

- Use up to 10 transaction or reference attributes as inputs into a mapping.
- Define default output value to use when actual input values don't match the mappings.
- Use wildcards for multiple input mapping sets to indicate that the value of a particular input should be ignored for certain mappings.
- Enter the mappings directly on the user interface or use the spreadsheet available in the Export option, and then import.

Export allows:

- Exporting a template to create new mappings.
- Exporting all mappings created for the mapping set to add or edit the current mappings.

Example

Assume a business operates in several regions, including:

- East
- South
- West

The business has a Region segment in their chart of accounts.

The region name can be the input for the mappings to derive the value of the region segment. You can create a mapping set that maps region names to the corresponding region code as described below.

Input Value (Region Name)	Segment Value
East	01
South	02



Input Value (Region Name)	Segment Value
West	03

Additional transaction information, such as transaction type and salesperson name, could also be used as inputs to help derive a different segment value for each combination of the input values.

Defining Mapping Sets: Examples

Define a mapping set when you have a matrix of input values that produces distinct output values. For each input value, specify a corresponding account combination or segment output value. One or more related pairs of these input values with the segment or account combination output values form a mapping set.

A mapping set definition includes the selection of input sources, output type, and mappings. The mappings section displays how input values are mapped to output values.

To define mapping sets:

- Specify the output type:
 - o The output type for a mapping set can be an account combination, segment, or value set.
 - Use value set:
 - If the value set is used by more than one chart of accounts,
 - And the mapping set can be reused across multiple charts of accounts.
 - Expected input or output combinations are constant across the charts of accounts.
 - Based on the selection, the mapping set provides the value for an account, segment, or value set.
- Define the input source:
 - Specify the input source for mapping.
 - The input source is provided for predefined mapping sets.
- Define the chart of accounts and value sets.
- Specify the output value for the mapping:
 - For a given input value, enter the corresponding output value.
 - The account rule uses this value to populate either the account or the segment.
 - If the output type is a value set, the output value is an individual value from the value set entered.
 - If the output type is segment, the output value is an individual segment value.
 - If the output type is account combination, the output value is an entire account.

Mapping sets are used with account rules:

- If the output type is account combination or segment, identify the chart of accounts assigned to the mapping set.
- If the output type is a value set, identify the value set assigned to the mapping set.
- If defining a mapping set for more than one chart of accounts or value set, it can be assigned to more than one account rule. This increases the ability to share the mapping set.



A mapping set with no associated chart of accounts:

- Can be assigned to an account rule if:
 - The account rule is not associated with a chart of accounts.
 - The mapping set can have any chart of accounts or no chart of accounts.
- Cannot be assigned to an account rule if:
 - The account rule is associated to a chart of accounts. The mapping set must have the same chart of accounts.

Example

In the following example, the chart of accounts is set up with four segments. A mapping set is defined with a value set for Supplier Type as described in the following table.

Input Value	Output Value
Services	01-100-6120-000
Consulting	01-400-6110-000

Assume that two invoices are entered, one for a supplier with a type of Services and one for a supplier with a type of Manufacturing.

When using the mapping set, the source value Supplier Type is compared with the mapping set input values to determine the account.

In this example, there is a match for the first case; the invoice with a supplier type of Services maps to an input value. However, the invoice with a supplier type of Manufacturing does not map to an input value.

The accounts are derived and described in the following table.

Invoice	Supplier Type	Output Value
1	Services	01-100-6120-000
2	Manufacturing	No account generated

Note: To ensure that transaction 2 is accounted, you may want to modify the account rule to which the mapping set is assigned. If not, a separate rule can be defined to provide for the Manufacturing supplier type, or define a default output in the existing mapping set.



Creating Description Rules: Explained

Use descriptions rules to define the elements of a description that appears on the subledger journal entry at the header or the line. The definition determines both the content and sequence in which the elements of the description appear. You can assign a condition to a description rule to determine that the description is selected for display if the condition is satisfied.

Description Rule Definition

A description rule can be defined with combinations of source and literal values. If sources are used in the rule, the accounting event class associated with the sources determines in which subledger journal entry rule set the description rule can be selected and used.

Build descriptions using the available sources for the application.

The following is the description details that have been entered, using a literal and a source:

- Loan Origination Date = Origination Date
 - Literal = Loan Origination Date
 - Source = Origination Date

For example:

- Source value of the Origination Date = 11/01/11
- Journal entry description = Loan Origination Date 11/01/11

Creating Conditions: Examples

The following provides examples of defining an account rule with a condition.

Example 1: Custom Real Estate Application Account Rule Condition Example

This example defines an account rule for assignment to a loan journal line. The account rule has two priorities, a mapping set and a constant.

- The first priority creates an output for an account based on the mapping set rule definition.
 - o A condition is created using the first priority rule. This rule is only used if the condition below is met.
 - The condition is Credit Status must not be null.
 - The accounts derived from the mapping set rule are used if the Credit Status has a valid value.
 Otherwise, the accounts derived from the entered constants value from the second priority are used.

The following table describes the setup of the condition on the first priority:

(Source	Operator	Value)
("Credit Status"	is not null)



The second priority creates an output from a constant value (0.9100030.50034206331.0.0.0). No condition is associated with the second priority.

Example 2: Oracle Fusion Assets Account Rule Condition Example

This example defines a rule for a capital purchase. The rule is applied if the distribution account cost center is the same as the liability account cost center, and the asset tracking option is Yes.

This condition can be expressed as:

Where Distribution Cost Center = Liability Cost Center and Asset Tracking option = Yes

The following tables describe the setup of the condition:

(Source	Delimiter	Segment	Operator	Value	Delimiter	Segment)	And Or
("Distribution Account"		"Cost Center"	=	"Liability Account"		"Cost Center")	'AND'
("Asset Flag"			=	Yes)	

The following two rows of data are used in the accounting event, to which the account rule and condition applies.

Account Rule Condition Example: Accounting Event Data

Account	Invoice 1	Invoice 2	Asset Flag
Distribution Account	02-640-2210-1234	01-780-6120-0000	Yes
Liability Account	01-640-2210-0000	02-782-2210-0000	Yes

In the Accounting Event Data table above, assume the cost center segment is the second segment. When the account rule with this condition is used the account rule is applied to derive the account of Invoice 1 only. For Invoice 2, (assets tracking option = Yes), the cost center for the Distribution and Liability accounts are not the same. Both conditions must be met in order for the rule to apply.

Note:

- When an account source is selected or entered, you must also select or enter a specific segment. Select All if the full account is required to be used in the condition instead of a specific segment.
- The condition uses the account source and distribution account, along with a segment that you must provide. In this example, the cost center segment is provided.

Creating a Custom Formula

Watch: This video tutorial shows you how to define a custom formula to return a value to be used to create subledger journal entries.



Manage Custom Formulas: Explained

You can use Subledger Accounting custom formulas to derive a value that is used in a journal entry, such as a journal amount or description.

Formula Usage and Assignments

A custom formula can be used to:

- Calculate a numeric value
- Derive an alphanumeric value
- Return a date value

It can also be used in any of the following accounting rule components:

- Journal line rule
- Account rule
- Mapping set
- Description rule
- Supporting reference

Formula Definition

Define custom formulas using sources for the selected event class. Predefined functions and conditions can also be used to derive the resulting value.

- Enter source in double quotes (").
- Enter constant values in single quotes (').
- Enter date values in the format YYYY-MON-DD.

Example 1

A custom formula has been defined to use in the description rule for the journal entry rule set for the Invoice event class.

Steps:

- 1. Define a formula:
 - o IF "Project Number" IS NOT NULL AND "Invoice Date" >= '2015-Jan-01'
 THEN Concatenate("Project Number", "Invoice Number")
 ELSE Concatenate("Supplier Name", "Invoice Number")
- 2. Use the formula in a description rule.
- 3. Assign the description rule to a journal entry rule set.

Results:

Accounting is created for two project invoices and one nonproject invoice.



Invoice	Invoice Details	Journal Entry Description
1	Invoice number = MA0024	ABC Inc.MA0024
	Invoice date = 2014-Dec-11	
	Supplier name = ABC Inc.	
	Project number = 12345	
2	Invoice number = MA0045	12345MA0045
	Invoice date = 2015-Jan-30	
	Supplier name = ABC Inc.	
	Project number = 12345	
3	Invoice number = MA0012	ABC Inc.MA0012
	Invoice date = 2015-Jan-15	
	Supplier name = ABC Inc.	

Example 2

A journal entry is recorded for the actual 401k funding every quarter. The monthly accrued amount entry is created for each month of the quarter using the following custom formula.

Note that the accrual entry is to be reversed at the beginning of the next month by the accrual reversal feature.

Steps:

- 1. Define a formula:
 - "Fund Balance" / 3 * "Month Number in the Quarter"
- 2. Assign the formula to the Entered Amount accounting attribute for the event class.
- 3. Use the formula as the Entered Amount for the journal line rule.
- **4.** Assign the journal line rule to the journal entry rule set.

Fund balance = 1,200 USD

Month	Results
1	Accrual amount for the first month in the quarter: 1200 / 3 * 1 = 400
2	Accrual amount for the second month in the quarter: 1200 / 3 * 2 = 800
3	Accrual amount for the third month in the quarter: 1200 /3 * 3 = 1200

Predefined Formula Functions

The following predefined functions are available to be used in custom formulas.



Type	Function	Description	Example
Alphanumeric	Concatenate	Concatenate two strings into one single string.	Concatenate('This is ', 'a test.') returns 'This is a test'.
Alphanumeric	Substring	Extract part of a string.	Substring('How are you?', 3,5) returns 'w are'.
Date	FirstDayOfYear	Return the first day of the calendar year based on the parameter.	FirstDayOfYear('2013-Jul-11') returns '01-Jan-2013' in date format.
Date	LastDayOfYear	Return the last day of the calendar year based on the parameter.	LastDayOfYear('2013-Jul-11') returns '31-Dec-2013' in date format.
Date	FirstDayOfMonth	Return the first day of the month based on the parameter.	FirstDayOfMonth('2013-Jul-11') returns '01-Jul-2013' in date format.
Date	LastDayOfMonth	Return the last day of the month based on the parameter.	LastDayOfMonth('2013-Jul-11') returns '31-Jul-2013' in date format.
Date	AddMonth	Return the date of specified number of months after the date in parameter.	AddMonth('2013-Jul-11', 2) returns 11-Sep-2013 in date format.
Numeric	YearNum	Return the year of a date source in 4-digit format.	YearNum('2012-Feb-28') returns the number 2012.
Numeric	MonthNum	Return the month of a date source in numeric format.	MonthNum('2012-Feb-28') returns the number 2.
Numeric	DayNum	Return the day of date source in numeric format.	DayNum('2012-Feb-28') returns the number 28.
Numeric	Round	Round a number to a specific decimal place.	Round(183.1123, 2) returns 183.11
			Round(183.1123, -1) returns 180
Numeric	RoundUp	Round up a number to integer.	RoundUp(0.01) returns 1.
			RoundUp(1.50) returns 2.
Numeric	RoundDown	Round down a number to	RoundDown(0. 01) returns 0.
		integer.	RoundDown(1. 50) returns 1.
Numeric	Power	Return the power of a number.	Power(3,2) returns 9.



Type	Function	Description	Example
Numeric	NumberOfGLPeriod	Return the number of nonadjustment accounting periods between two dates. • If start date is not the first day of an accounting period it's counted as one period. • If end date is not the last day of an accounting period, it's counted as one period, it's counted as one period.	If accounting calendar is defined as Monthly, NumberOfGLPeriod('2013-Jul-11', '2013-Oct-10') returns 4.

Related Topics

- Subledger Journal Entry: Overview
- Creating a Manual Subledger Journal: Points to Consider
- Accrual Reversals: Explained

Creating Supporting References: Explained

Supporting references are used to store additional source information about a subledger journal entry at the line level.

Supporting references with balances establish subledger balances for a particular source and account for a particular combination of supporting references plus the account combination.

For example:

- If a journal line contains a supporting reference that includes two sources, Customer Type and Customer Name.
- Balances are created for the account combination, plus customer name and customer type.

Examples of how you may want to use supporting reference balances are to:

- Facilitate reconciliation back to the subledgers and source systems by tagging journal entries with transaction and reference attributes.
- Create balances by dimensions not captured in the chart of accounts.
- Reporting using dimensions not captured in the chart of accounts.
- Enrich Oracle Fusion Business Intelligence Applications reporting on subledger journals.
- Profit and loss balances by dimensions not captured in the chart of accounts

Define supporting references to hold additional supporting information for detailed account balance maintenance or reconciliation and reporting requirements.

Supporting Reference Assignment

Supporting references are a powerful tool to allow capture of journal entries with transaction attributes. You can use these tags to report on entries, reconcile entries back to source systems or even maintain balances at the attribute level.

 Define supporting references once and reuse by assigning sources of different event classes or source systems to the same supporting reference.



 You can assign one source per event class to each supporting reference. The subledger or source system uses the supporting reference name to store the source values. This standardizes supporting reference information, even if it comes from disparate source systems.

Supporting references can be defined using either of these options (located on tabs):

- With Balances:
 - Select the balances option in the definition of the supporting reference, to have balances only maintained when the supporting reference is assigned.
 - If balances are maintained for a supporting reference, they are carried forward into the next fiscal year, for all Profit and Loss account types.
 - A limit of thirty supporting references with balances can be defined. You can consider adding more source assignments to predefined supporting references, rather than creating a new one.
- Without Balances:
 - o No limit to the number of supporting references without balances is defined.
 - o Consider using a journal entry header or line description if:
 - No balance is maintained for a supporting reference.
 - No supporting reference details are needed for reports.

▲ Caution: Using supporting references instead of descriptions may impact accounting engine performance.

Related Topics

• Supporting Reference Assignments: Points to Consider

Migrate the Configuration

Migrating Accounting Rules: Points to Consider

Use the export and import functionality in the Setup and Maintenance work area to perform migration of setup data. When migrating accounting rules, you must migrate task lists in entirety and fulfill some requirements.

Full Task List Migration

This table shows the task lists to migrate in full, depending on the offering.

Offering	Task List	
Oracle Fusion Accounting Hub	Define Accounting Transformation Configuration	
Oracle Fusion Financials	Define Subledger Accounting Rules Define Transaction Account Rules	

Support is not provided for a partial task list migration.



You may migrate setup data for specific applications only. Note that supporting references, accounting options, and accounting class usages are migrated for all applications, regardless of the applications specified.

Migration Requirements

Prior to migration, journal entry rule sets and accounting methods must be successfully activated. Invalid journal entry rule sets or accounting methods cause import failure.

Ensure that your setup data migration includes all dependent business objects from other required setup modules, such as Define Ledgers. The import sequencing of these dependent business objects must be prior to accounting rules business objects.

Related Topics

- Configuration Packages: Explained
- Implementation Project Based Export and Import: Explained

Secure Accounting Transformations

Security for Accounting Transformations: Explained

Accounting transformations require both function and data security privileges.

Oracle Accounting Hub security for accounting transformations includes:

- Setup task security
 - Security to integrate your external systems with accounting transformations, indicating what types of transactions or activities require accounting from those systems.
 - Security to configure accounting rules to define accounting treatments for transactions.
- Transactional task security
 - Security to create subledger journal entries (manual subledger journal entries or those generated by the Create Accounting process).
 - Security to review and generate reports of subledger journal entry headers and lines.

Security to Perform Setup Tasks

Use the Define Accounting Transformation Configuration task in the Setup and Maintenance work area to integrate your external systems with the Accounting Hub.

To register your external systems and configure accounting rules, the setup user must be provisioned with a role that includes the Accounting Hub Administration Duty role.

- In the security reference implementation, the Financial Application Administrator job role hierarchy includes the Accounting Hub Administration Duty role. This role provides the access to integrate your external systems with accounting transformations.
- For more information on available setup job roles, duty roles and privileges, see the Oracle Fusion Accounting Hub Security Reference Manual.



Security to Perform Transactional Tasks

To create and view subledger journal entries, you must have the access necessary to perform the tasks. These tasks can be accessed from the General Ledger, Journals work area. You must have access to the work area, and the ledgers in which the journal entry is posted.

The following are defined in the security reference implementation:

- The General Accounting Manager job role hierarchy includes duty roles that provide entitlement to manage your general accounting functions. This entitlement provides access to the General Ledger Journals work area.
- The General Accounting Manager role hierarchy includes data security policies that provide entitlements to access ledger and subledger journal entries.
 - Ledger access is provided through Data Access Sets.

The following duty roles must be assigned directly to the General Accounting Manager job role. This provides access to create and view subledger journal entries:

- Subledger Accounting Manager Duty
- Subledger Accounting Reporting Duty

Alternatively, you can assign the Subledger Accounting Duty and Subledger Accounting Reporting Duty roles to any of the following General Ledger job roles:

- Financial Analyst
- General Accountant

Related Topics

Data Security: Explained

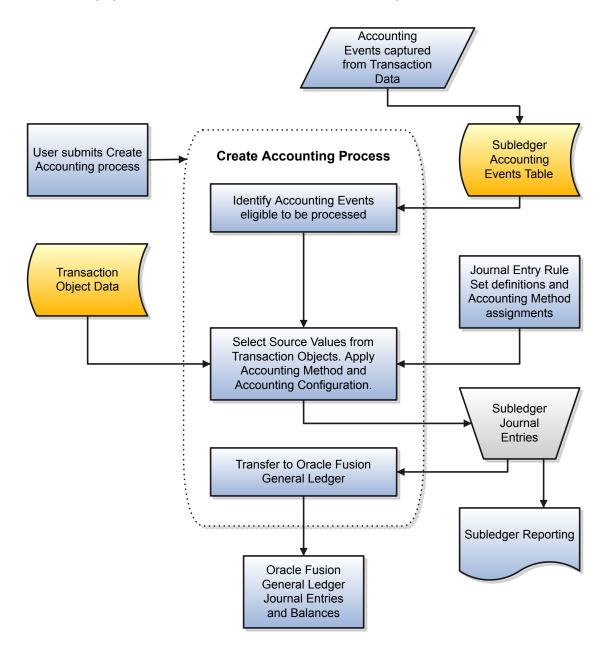
Create and Process Subledger Journal Entries

Subledger Journal Entries: How They Are Created and Processed

The Create Accounting process uses the transaction objects data to create subledger journal entries. For example, if a subledger journal entry rule set specifies that the customer name should appear in the description of a subledger journal entry line, then the customer name value is taken from the customer name source data provided by the transaction objects.



The following figure illustrates the process used to create subledger journal entries.



How Subledger Journal Entries Are Created and Processed

- 1. When transactions are committed in a subledger, accounting events are captured and stored in the subledger accounting events table.
- 2. The Create Accounting process identifies all accounting events eligible to be processed. For each of these events, the transaction objects provide the Create Accounting process with the transaction objects data (source information). This is the contextual data of the transaction, such as amounts and accounting dates.
- 3. When the Create Accounting process is run, subledger journal entry rule set definitions and transaction objects data are applied to the transaction object data to create subledger journal entries.



4. These entries are summarized and transferred to Oracle Fusion General Ledger.



8 Manage Subledger Accounting

Manage Subledger Accounting: Overview

In the Manage Subledger Accounting activity, you can generate journal entries for source system transactions, create adjustment entries, and review accounting results.

You can:

- Create accounting for a batch of transactions by submitting an offline process.
- · Create manual adjustment entries.
- Review generated journal entries and projected balances on views and reports.

Create Accounting

Subledger Accounting Options: Explained

Subledger accounting options define how certain accounting processing should be done for transactions of a given subledger at a ledger level. These options are set up for the primary and secondary ledgers only.

Manage Subledger Accounting Options

This task is accessed from the Setup and Maintenance work area. The page is displayed in the context of a primary or secondary ledger. All registered subledger applications are displayed. If you created additional accounting event classes after initial ledger setup, run the Update Subledger Accounting Options process to incorporate these event classes.

The Manage Subledger Accounting Options task provides the ability to edit:

- Accounting Options
- System Options

Edit Accounting Options

This page displays the subledger accounting options for the selected ledger. You can review and update the options. The view for this page depends on the subledger application type and the ledger type (primary or secondary).

The views are as follows:

- The Defaults and Ledger options view displays the accounting program defaults and the event class options for the following:
 - o A primary ledger for a subledger application.
 - A secondary ledger for a subledger application.
- The Ledger options view displays the event class options for a subledger application, and secondary ledger.



General Options

Subledger Accounting Enabled

This option is visible only for secondary ledgers in the ledger options view. Enable or disable the subledger application for the ledger. No entries are generated by subledger accounting for an application if subledger accounting is disabled for the ledger.

- General Ledger Journal Entry Summarization
 - This option determines whether subledger journal entries are summarized or grouped when they are transferred to General Ledger as described in the table below.

Option	Description		
Summarize by general ledger period	Default option.		
	Indicates that all subledger entry lines are summarized into a single general ledger entry if they have the same:		
	o Accounting period		
	o General ledger journal category		
	o Account		
	o Entered currency		
	o Side		
	o Balance type		
	Disabled if the ledger uses daily balancing.		
	The general ledger effective date defaults to the last date of the accounting period.		
Summarize by general ledger date	Indicates that all subledger entry lines are summarized into a general ledger entry if they have the same:		
	o Accounting date		
	o General ledger journal category		
	o Account		
	o Entered currency		
	o Side		
	o Balance type		
	Default value if Summarize by accounting period is disabled.		
	The general ledger effective date is equal to the subledger accounting date.		
Group by general ledger period	Indicates that all subledger journal entries with the same accounting period are grouped together.		
	Each subledger journal entry line is transferred into the general ledger with the same granularity as the original lines.		



Option	Description
Group by general ledger date	Indicates that all subledger journal entries with the same accounting date are grouped together.
	Each subledger journal entry line is transferred into the general ledger with the same granularity as the original lines.

Reversal Method

Use the Reversal Method option to determine how the reversal subledger journal entries are generated in subledger accounting. The options are:

- Switch debit and credit (default): reverses a debit for a credit and a credit for a debit.
- o Change Sign: The reversal entry keeps the same side as the original entry, but the sign is changed.

Rounding Rule

Use the Rounding Rule option to determine which option to use for rounding. Subledger accounting rounds to the minimum accountable unit or ledger currency precision. The table below describes the rounding rule options.

▼ Tip: For examples described in the Rounding Rule Options table below, assume that the precision is 2 and the minimum account unit is 0.01.

Option	Description		
Up	Amount rounded up. For example, \$5.983 is rounded to \$5.99.		
Down	Amount rounded down. For example, \$5.988 is rounded to \$5.98.		
Nearest	Amount rounded to nearest number. However, if the difference between Up and Down options is equal, the amount is rounded up. For example, \$5.985 is rounded to \$5.99.		

Third-Party Merge Accounting Options

The reporting currency ledgers inherit the third-party merge accounting option from the primary ledger. The table below describes the third-party merge accounting options.

Option	Description		
Transfer of the third-party control account balances	 Transfers the control account balance at the merge date from the old third party to the new third party. Reverses and rebooks the existing journal entries that occurred after the merge date. 		
	 In the case of a partial merge, transfers the balance that corresponds to the transactions that are part of the partial merge. 		
Replace third party	 Updates the existing journal entries by replacing the old third party and site with the new third party and site. 		
None	o No accounting effect.		



Journal Categories Options

 The table below describes the event class option that can be overridden. Note that only the Journal Category event class can be overridden.

Option	Description
Journal Category	Defined in the accounting rules for an event class. Any valid journal category defined in General Ledger can be selected.

Edit System Options

The following three regions are available:

- Create Accounting Processing Options
- Event Class Options
- Transfer to General Ledger Processing Options

Create Accounting Processing Options

- Number of Create Accounting Workers
- Processing Unit Size

Column Name	Description
Processing Unit Size	Approximate batch size for high volume general ledger transfer flow, and is also the general ledger import batch size. If not specified, the corresponding accounting processing unit size is used.

- You can define the processing unit size to process a large number of items in one commit cycle. A processing unit is the number of transactions processed by the Create Accounting process in one commit cycle. Create Accounting processes the default processing unit size at the application level.
- Stop at Error Limit
 - o Maximum number of events allowed to fail before canceling account program.

Event Class Options

- Event Class
- Processing Unit Size

On the **Manage Subledger Accounting Options** page, you can select the subledgers for each ledger or ledger set, and specify the **Processing Unit Size** for each event class.

Transfer to General Ledger Processing Options

- Number of General Ledger Transfer Workers
- Processing Unit Size

Importing information from subledgers is done using subledger accounting. Posting from the subledger systems transfers data to the general ledger interface and journal entry tables.



As part of your configuration, you can specify whether the Create Accounting process is to split the creation process into multiple workers (processors). The benefit of splitting the creation process is that:

- Accounting can be generated more efficiently with more resources allocated to complete the process.
- You can have multiple processors running in parallel to create the entries in a shorter period of time.

One restriction is the capacity of the hardware that is being used by the application. The more available processors, the more you are able to allocate to the Create Accounting process.

The decision for how many processors to use is made based upon expected volumes and the processing window. In other words, how much time is allocated to creating accounting. Accounting is often done as a nightly batch process, to facilitate system availability during work hours.

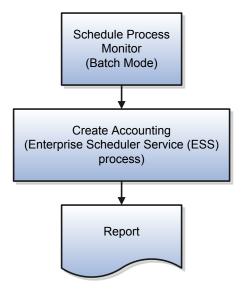
Dependencies exist between the overall completion status of the Create Accounting process and the workers. In general, the parent does not update to the Completed status until all the workers successfully complete.

The process that allows transfer of subledger journal entries to general ledger uses separate processing workers that are specialized in general ledger transfer. A lightweight general ledger transfer parent process is used to distribute the workload to the workers. To transfer entries even faster, you can have a number of parallel processing workers used for high volume general ledger transfer flow. If not specified, the corresponding accounting processors are used.

Submitting the Create Accounting Process: Explained

The Create Accounting process is an Enterprise Scheduler Service (ESS) process. It can be submitted as a request from the Scheduled Processes Monitor window to create journal entries for a batch of events. It has input parameters that determine the execution mode and the selection criteria for events.

The figure below shows the submission of the Create Accounting process.



The following table describes the parameters for the Create Accounting process as submitted in the Scheduled Processes Monitor window.



Prompt	Description		
Subledger Application	Source system for which the Create Accounting process is being executed.		
Ledger	Ledger name for which the Create Accounting process is being executed.		
Process Category	Selecting a process category indicates that all associated accounting event classes and their accounting event types are selected for processing.		
End Date	End date puts a filter on the selection of events. Only events having an event date on or before the end date are selected for accounting.		
	Default value is current system date. If the process is scheduled to execute periodically, after the initial process, the End Date for each subsequent scheduled process is incremental.		
Accounting Mode	Accounting mode; Draft or Final		
	Default value is Final.		
Process Events	Adds other filter criteria for the Create Accounting process to select events:		
	All: Process all events.		
	Errors: Process only those events that have previously been processed in error.		
	Invalid Accounts: Process only those events that have previously been processed in error. Replace any invalid accounts with the suspense account.		
	Default value is All.		
Report Style	Users can select the type of details to display on the execution report. The report can be printed in Summary, Detail, or No report.		
	Default value is Summary.		
Transfer to General Ledger	Indicates whether the Create Accounting process should submit the Transfer to GL process; Yes, No.		
	Default value is Yes.		
Post in General Ledger	Indicates if users, who have been granted the posting privilege, want to submit General Ledger posting; Yes or No.		
	Default value is Yes.		
Journal Batch	Batch name used by Transfer to GL to decide on the batch name for the batch created in Oracle Fusion General Ledger.		
	When a value for the batch name is not provided, journal import defaults will be used.		



Prompt	Description	
	This is a free text field.	
Include User Transaction Identifiers	Default value is No.	

Related Topics

• Disabling Posting Subledger Transactions to the General Ledger: Explained

Accessing the Create Accounting Execution Report: Explained

When you submit the Create Accounting process, the Create Accounting Execution Report is submitted automatically if the Create Accounting process completes with a success or warning status. The Create Accounting process output contains a message with the Create Accounting Execution Report request identifier. Use this request identifier to retrieve the execution report output.

Create Accounting Execution Report

The Create Accounting Execution Report can be recreated.

This report can be recreated as needed using the following criteria:

- Submit the Create Accounting process.
- Use the same request identifier of the wanted Create Accounting process run previously by the same user, provided that no other Create Accounting process was run with the same criteria.

This report can be recreated as needed using the following criteria:

If you select to transfer the entries to Oracle Fusion General Ledger when submitting the Create Accounting process, the report indicates if accounting entries have been transferred or not transferred.

Related Topics

Parameter Specifications: Explained

Diagnose Subledger Accounting Event Data: Explained

The diagnostic framework stores data that is used in the creation of a journal entry so that the data can be reported to analyze accounting issues. The purpose of the process is to provide the transaction data that is referenced during accounting through accounting rules and ledger setup.

The diagnostic framework provides a tool that can be used to determine why the accounting was created in a particular way or why there are errors. Errors may occur because either the source system information or the accounting rules are not as expected.

The following examples describe typical uses of the diagnostic framework features.

• In the implementation phase, you can launch the Accounting Event Diagnostic report to review the source values available in the transaction objects.



On a daily basis, you can use the Accounting Event Diagnostic report to troubleshoot exceptions.

Diagnostic Framework Features

The diagnostic framework features are as follows:

- SLA: Diagnostic Enabled: This option controls whether diagnostic information is gathered by the Create Accounting process. Use with caution. Selecting this option can cause slow performance.
- Diagnostic Framework Execution: When the SLA: Diagnostic Enabled option is set to Yes, the diagnostic framework is executed simultaneously with the Create Accounting process. The diagnostic framework data is stored in the diagnostic tables.
- Accounting Event Diagnostic Report: To view the diagnostic framework report, users submit the Accounting Event Diagnostic process with the appropriate report parameters.
- Purge Accounting Event Diagnostic Data: Purging is useful when accounting is successfully created so that the volume of information in the report does not increase to a point where it becomes unusable.

Diagnostic framework data purged:

- When the data collected can be purged by running the process.
- When the administrator launches the Purge Accounting Event Diagnostic Data process.

Diagnostic Framework Business Process Flow

The following steps describe the diagnostic framework business process flow:

- 1. The administrator sets the SLA: Diagnostics Enabled option to Yes for the user or responsibility.
- 2. Users submit the Create Accounting process that automatically executes the diagnostic framework to populate the diagnostic tables. The diagnostic framework gathers source values and other information from the transaction objects.
- 3. Users submit the Accounting Event Diagnostic process to view the diagnostic report. Based on the information in the diagnostic report, users may determine if additional or resubmission of information from source systems is required. They may also determine if any updates to the accounting rules is required.
- 4. Users run the Create Accounting process again to generate subledger journal entries for events that had an error.
- **5.** The administrator submits the Purge Accounting Event Diagnostic Data process to purge the results of the diagnostic framework.

Posting Subledger Transactions to the General Ledger: Explained

Post subledger journals to the general ledger when you create accounting, or run the Post Subledger Journal Entries process to transfer and post at a later time.

To perform posting to General Ledger, you must have the privilege to execute this task.

With the required privileges, you can select the Account and Post to Ledger option to create and post journal entries to the general ledger, during the online Create Accounting process.

- If you do not have the privilege to post, select Account in Final to create journal entries and transfer to General Ledger.
- You can view the output of the Post Subledger Journal Entries process for the summary of the transfer process.



You can manually run the Create Accounting process to create a batch of transactions. Set the following parameters to create journal entries and post to the general ledger.

- Accounting Mode is set to Final.
- Transfer to General Ledger is set to Yes.
- Post in General Ledger is set to Yes.
 - If you do not have the privilege to post, the Post in General Ledger parameter is not available.

Related Topics

• Disabling Posting Subledger Transactions to the General Ledger: Explained

Create Subledger Journal Adjustment

Subledger Journal Adjustments: Points to Consider

Subledger journal adjustments enable easier audit and reconciliation because you can store transaction information directly with journal adjustments. You can enter and complete subledger journal adjustment entries from the user interface or upload using a spreadsheet.

You can perform the following actions on your subledger journal adjustments, depending on their status:

- Edit
- Duplicate
- Reverse
- Delete
- Complete

Editing a Subledger Journal Adjustment

When editing a subledger journal adjustment, you can perform the following tasks:

- Edit the journal header information, if the status is not Final.
- Edit and create journal lines, including accounts.
- Enter the debit, and credit amounts.
- Enter the accounting class.
- View the impact on general ledger account balances should the adjustment be completed.
- Post the journal.

You can also edit incomplete subledger journal adjustments.

- Examples of header information which can be updated:
 - Ledger
 - Accounting date



- Category
- Description
- Examples of subledger journal adjustment line information which can be updated:
 - Account
 - Accounting class
 - Entered amount
 - Journal adjustment lines
 - Select supporting references and assign values to them.
- Edit default currency options to be assigned to a subledger journal adjustment.
- Edit or redefine the subledger journal adjustment description.

Duplicating Subledger Journal Adjustments

As a time saving feature, you may copy an existing adjustment.

The duplication feature is enabled for all existing subledger journal adjustments, regardless of status. It includes the ability to copy complete information required for a subledger journal adjustment header and line.

Note: All fields can be edited when an adjustment is duplicated.

Reversing Subledger Journal Adjustments

You can reverse subledger journal adjustments in Final status. Reversal options are populated from accounting options.

- Switch debit and credit.
- Change sign.

Deleting Subledger Journal Adjustments

Oracle Fusion Subledger Accounting provides the ability to delete a subledger journal adjustment that is not in Final status. The ability to delete subledger journal adjustments ensures that users have the flexibility to correct errors without technical support.

Completing Subledger Journal Adjustments

You can complete subledger journal adjustments in Final and Post to General Ledger status.

Creating a Manual Subledger Journal: Points to Consider

The application enables the user to create manual subledger journal entries online.

Creating a Manual Subledger Journal Entry

This includes the ability to:

- Enter the complete information required for a manual subledger journal entry.
- Enter subledger journal entry descriptions.
- Select a supporting reference and supply the supporting reference value to a subledger journal entry line.



- Assign a descriptive flexfield to a subledger journal entry header or subledger journal entry line.
- Populate default values for an entered currency for a created subledger journal entry line.
- Enter default conversion type, date, and rate information to establish a default currency for the journal that is different than its associated ledger currency.
- View projected balances of entered and accounted journal line amounts.
- Complete and post subledger journal entries.

Note: The ability to post subledger journals to the General Ledger is dependent on your security profile. If you do not have the privilege to post, creating a manual subledger journal entry with a Final completion status includes the transfer to General Ledger.

Supporting Reference Assignments: Points to Consider

You may want to analyze account balances and financial results by different transaction attributes. However, transaction information such as salesperson, customer, or supplier are typically not stored in the general ledger because of the volume of general ledger balances it would create. Therefore, you are not able to analyze general ledger data categorized by transaction attributes.

You can perform this type of reporting and analysis using supporting reference information from subledger transactions. This feature enables you to create balances based upon transaction attributes not typically stored as segments in the general ledger chart of accounts.

For example, you can report on receivables balances by salesperson, customer, credit risk category, item, or any other combination of transaction attributes.

Supporting references can be used to:

- Provide additional information about a subledger journal entry line.
- Establish a subledger balance for a particular supporting reference value.
- Assist with reconciliation of account balances.
- Provide additional detail information for financial managerial analysis.

You can assign supporting references at the subledger journal entry line level.

Assigning Supporting References at the Subledger Journal Entry Line

Assigning supporting references to subledger journal entry lines enables you to maintain detailed balances, by supporting reference value, for general ledger accounts.

Related Topics

Creating Supporting References: Explained

Validating a Third-Party Control Account: Examples

If third-party control accounts are enabled, and the account entered is a third-party control account, you must enter third-party information in the journal entry.



Example

If an account is defined as a Supplier type third-party control account the subledger journal entry lines that use that account must include supplier information.

When a valid third-party control account is assigned to a journal line, you are required to provide third-party information, such as name and site.

Submit the Third-Party Balances Report to display subledger balances and account activity information for suppliers and customers.

The Customer or Supplier subledger third-party balances are updated when the journal is completed to a Final status.

Review Subledger Journal Entry

Subledger Journal Entry: Overview

You can create subledger journal entries by using one of two methods:

- Use the Create Accounting process to process accounting events using accounting rules.
- Create manual subledger journal entries.

Subledger journal entries are always created in a given subledger application context. When the subledger journal entry is complete, the appropriate sequence names and numbers are assigned. If applicable, corresponding secondary ledger and reporting currency journal entries are also created.

Manual journal entries can be entered for primary ledgers or for secondary ledgers. Manual journals for primary ledgers are automatically created only for associated reporting currencies, not secondary ledgers.

Reviewing a Subledger Journal Entry: Points to Consider

Subledger journal entries can be reviewed whether they were created from processing accounting events, or manually entered.

You can search subledger journal entries directly, or search for journal entries with a specific status, unprocessed accounting events, or subledger journal entry lines.

Advanced search functionality, including the ability to use multiple search criteria is available.

Review Subledger Journal Entries

Perform an inquiry on unprocessed accounting events, subledger journal entries and subledger journal entry lines based on multiple selection criteria.

- Create, edit, duplicate, reverse or delete a manual subledger journal entry
- View detailed information about the subledger journal entry
- View a subledger journal entry in the T-Accounts format
- View transactions underlying the accounting event or the subledger journal entry



- · View supporting references associated with the subledger journal entry and lines
- View tagged subledger journal entries or create a tag on the subledger journal entry
- Perform subledger journal corrections by editing the account directly on a completed journal entry. Review the original entry together with the correction for easier reconciliation.
 - o To correct the account on a subledger journal you must have the Override Subledger Journal Entry privilege.

Viewing Projected Balances: Points to Consider

Use the projected balances feature to view the impact on account balances for selected subledger journal entry lines.

The projected balances flow has the following business benefits:

- The creation and validation of unposted manual journal entries by providing immediate information about the account balances for the selected journal lines.
- The validation and reconciliation of posted journal entries by providing immediate information about the account balances for the selected journal lines.

Projected Balances

Subledger Accounting manual journal entry and General Ledger manual journal entry, approval, and inquiry pages display projected or current balances including the current journal entry line amounts. Depending on whether the journal is posted or not, the current balance (for the period of the journal) is displayed or calculated.

The projected balance region displays the projected balances for the account that includes the amounts of the selected journal entry line. Additionally, if more than one journal line for same account of the selected journal line exists, then the projected balance amount includes the impact from each journal line. The Period To Date, Quarter To Date, and Year To Date balances are also available.

- For unposted journals, the period balance is projected by summing the current balance with the subledger journal entry line amounts
- For posted journals, the opening balance and the period activity are calculated using the current balance and journal line amount

Projected balances increases accuracy when booking entries to reconcile accounts.

Managing Accounting Reversals: Explained

To create an accounting reversal for a transaction or transaction distribution, the transaction objects should include the appropriate header or line level accounting reversal options.

Accounting reversals enables you to reverse the accounting impact of a previously accounted transaction distribution or all existing accounting for a transaction.

Accounting reversal terminology includes the following:

- · Reversed (original) Distribution
 - Refers to a transaction distribution that although successfully accounted, is either incorrect or canceled. The transaction distribution is therefore reversed.



- Reversal Distribution
 - Refers to a transaction distribution which reverses the effect of the original distribution on transaction balances. Typically, reversal distributions are identical to the reversed distributions in all respects except for entered (ledger) amounts that reverse the sign of the original.
- Replacement Distribution
 - o Refers to a transaction distribution which replaces the reversed distribution with the correct value.

Distribution Examples

The table below illustrates the distributions described above.

Invoice Distribution Line Number	Invoice Line Type	Accounting Date	Amount	Description
1	Item	10-Jan-2013	1000	Reversed
2	Item	12-Jan-2013	-1000	Reversal (of line 1)
3	Item	12-Jan-2013	2000	Replacement (of line 1)
Transaction Total			2000	

Note that the original accounting impact of the reversed distributions is undone, even if the subledger journal set ups or accounting configurations have changed since the original subledger journal entry was generated.

Subledger Balances

Import Supporting Reference Initial Balances

Use the Import Supporting Reference Initial Balances task to import initial supporting references balances. This task enables you to upload subledger journal entries to initialize supporting reference balances. The uploaded journal entries don't impact general ledger balances.

Note: When using the Manage Configuration Packages task to export or import setup data, initial balances that have been imported into one environment are not going to be migrated into another. You must import balances separately into each environment you maintain.

Access the task from the Setup and Maintenance work area.



Third-Party Control Account Balances Report

Use the Third-Party Control Account Balances report (XLATPRPT) to list subledger accounting third-party control account balances and activity for suppliers and customers.

The report provides the following information:

- Third-party balances for subledger accounting third-party control accounts.
- Subledger journal entry lines that add up to the total period activity for each control account, third party, and third-party site.
- Third party and third-party site information.
- User transaction identifiers for the associated events.

Run the process from the Scheduled Processes page and optionally schedule the process to run periodically.

Parameters

Journal Source

List of enabled journal sources mapped to registered accounting generating subledger applications.

Ledger

The list of ledgers is based on the user accessible business units, asset books or legal entities. Only the list of ledgers associated with those business units, asset books or legal entities is shown.

From Accounting Period

All periods for the accounting calendar and period type associated with the selected ledger.

To Accounting Period

All periods for the accounting calendar and period type associated with the selected ledger.

Third-Party Type

Based on third party defined for the subledger application for the journal source. Values: Customer, Supplier, All.

Third-Party Name

All customers or suppliers based on third party type.

Third-Party Site

All third-party sites for the supplier or customer identified by third-party name.

From Third-Party Number

All customer or all supplier numbers based on the third party type.

To Third-Party Number



Any customer or supplier numbers greater than or equal to the From value.

Include User Transaction Identifier

Controls whether the report retrieves user transaction identifiers names and values. Values: Yes, No.

Account

Account range filter.

Related Topics

- · Accounting Attribute Assignments: Points to Consider
- Subledger Accounting Reports: Overview

Update Subledger Accounting Balances Process

Use the Update Subledger Accounting Balances process (XLABABUP) to update subledger supporting reference balances and third-party control account balances.

This process is run automatically following the Create Accounting process. It should be run manually if the auto run fails for any reason.

Run the report from the Scheduled Processes page and optionally schedule the process to run periodically.

Note: Draft entries do not update balances. To include accurate balances, transactions must be accounted in Final.

Parameters

Subledger Application

List of all user accessible accounting generating subledgers.

Ledger

The list of ledgers is based on the user accessible business units, asset books or legal entities. Only the list of ledgers associated with those business units, asset books or legal entities is shown.

Carry Forward Subledger Accounting Balances Process

The Carry Forward Subledger Accounting Balances process will carry forward supporting reference balances and third-party control account balances to the current period for a ledger. This process is automatically initiated when the accounting period is newly opened.

Note: The process can also be submitted manually if the initial process executed abnormally when the accounting period was opened.



Parameters

Ledger

The list of ledgers is based on the business units, asset books or legal entities that a subledger user has access to. Only the list of ledgers associated with those business units, asset books or legal entities is shown.

Accounting Period

All periods for the accounting calendar and period type associated with the selected ledger.

Advanced Features

Accounting Class Usages: Explained

Accounting class usages is a classification or grouping of accounting event classes to be used in a report or process based on subledger journal entries. A report or process referring to such a group would only process subledger journal entries tagged with the accounting event classes defined in the group.

Manage Accounting Class Usages Task

The Manage Accounting Class Usages task enables you to group accounting event classes. Use Accounting Class Usages assignments to determine which subledger journal entry lines to retrieve for a particular process. For example, for mass additions a subledger can define accounting class usages to identify the journal entry lines that must be processed to create records for another subledger.

Defining Accounting Class Usages

Use the Manage Accounting Class Usages task to create assignments that may be associated with a ledger. In the Accounting Class Assignments region, assign accounting classes to an assignment definition.

Predefined processes and assignment definitions cannot be deleted or updated. You can copy a predefined assignment definition and modify if necessary.

Accrual Reversals: Explained

To use accrual reversal, assign application or standard date sources to the Accrual Reversal Accounting Date Source accounting attribute. When this accounting attribute returns a value, the Create Accounting process generates an entry that reverses the accrual entry.

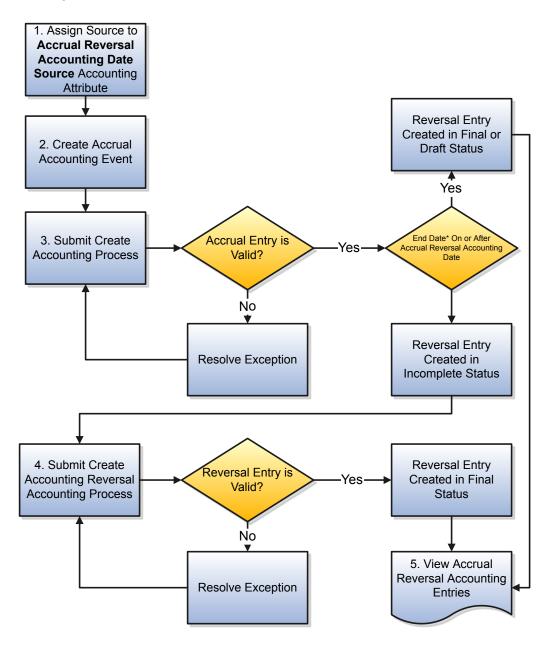
You can define how and when accrual reversals are automatically performed by:

- Indicating that an accounting event is eligible for accrual reversal.
- Determining when the accrual is reversed.
- Scheduling the Create Accrual Reversal Accounting process to generate the reversal entries of the accrual.



Accrual Reversal Process Steps

The diagram below displays the accrual reversal process:



Note: * End Date: The date up to which the Create Accounting process selects accounting events for processing.

Accrual Reversal Process Steps

1. Assign a date source to the Accrual Reversal Accounting Date Source accounting attribute at the event class level.



Use this attribute to schedule the automatic reversal of a journal entry. Assign any standard date source or one of the following application sources to the Accrual Reversal Accounting Date Source accounting attribute:

- Next Day: The accounting date of the accrual reversal is the next day following the accounting date of the accrual entry.
- First Day of Next Accounting Period: The accounting date of the accrual reversal entry is the first day of the following accounting period.
- Last Day of Next Accounting Period: The accounting date of the accrual reversal entry is the last day of the following accounting period.
- Note: You can override the Accrual Reversal Accounting Date Source accounting attribute values on the journal entry rule set if multiple sources have been assigned to the accounting attribute.
- 2. Create an accounting event in the subledger application.
- 3. Submit the Create Accounting process.

The Create Accounting process creates the accrual journal entry as well as the accrual reversal journal entry. The process creates the accrual reversal entry to negate the impact of the accrual entry. The Manage Subledger Accounting Options, Reversal Method option enables entries to appear as debit and credit signs reversed or to have negative amounts.

The status of the accrual reversal journal entry is based upon both the:

- Accounting period status of the accrual reversal journal entry accounting date.
- End Date specified in the Create Accounting process parameters.

Accounting Date	Open or Future Entry Accounting Period	Closed or Permanently Closed Accounting Period	Never Opened Accounting Period
Accounting date of the accrual reversal is on or before the end date specified.	Creates the journal entry with the accounting mode specified.	 Adjusts the accounting date to the next open or future accounting period. Creates the journal entry with an adjusted accounting date if the adjusted accounting date is on or before the end date. Creates the journal entry with an adjusted accounting date in an Incomplete status, if the adjusted accounting date is after the end date. If no open or future accounting period is found, creates the journal entry with an Error status. 	Creates the journal entry with an Error status.
Accounting date of the accrual reversal is after the end date specified.	Creates the journal entry with an Incomplete status.	 Adjusts the accounting date to the next open or future accounting period. Creates the journal entry with an adjusted accounting date in an Incomplete status. 	Creates the journal entry with an Incomplete status.



Accounting Date	Open or Future Entry	Closed or Permanently Closed	Never Opened Accounting
	Accounting Period	Accounting Period	Period
		 If no open accounting period is found, creates the journal entry with an Error status. 	

- 4. Submit the Create Accrual Reversal Accounting process to complete accrual reversals in future accounting periods as the periods are opened.
- 5. View accrual reversal accounting entries.

Related Topics

• Accounting Attribute Assignments: Points to Consider

Accrual Reversal: Examples

The following examples describe how accrual reversals are scheduled and accounted.

Example 1

A company receives materials worth 100 (USD) on the 30th of the month but has not been invoiced. The following journal entry is created when the material is received to record the accrual.

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Accrual Expense	100.00		100.00	
Accrual Liability		100.00		100.00

The accrual reversal accounting date is set to First Day of Next Accounting Period. The following journal entry is created to reverse the accrual.

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Accrual Liability	100.00		100.00	
Accrual Expense		100.00		100.00

Example 2

Future trading requires a margin account that is market-to-market on a daily basis. This means that the investor or ledger's gains or losses on the position are reflected on a daily basis. If the margin account drops below a specified amount (the maintenance of the margin), a margin call is issued. This requires the holder of the account to replenish the account to the initial margin level or close out the position.

The investor or ledger must mark the account to market each day and the entry booked from the day before must be reversed to reflect the new position.



Journal Entry Creation

- The following journal entry creation date is June 1, 2006.
- Accounting date: 01-Jun-2006

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Loss	100.00		100.00	
Margin Liability		100.00		100.00

Reverse the Journal Entry

- The accrual reversal accounting date is set to Next Day. The following journal entry is created to reverse the journal entry from June 1.
- Accounting Date: 02-Jun-2006

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Margin Liability	100.00		100.00	
Loss		100.00		100.00

Reflect the New Position

- On June 2, a new journal entry is created to reflect the new position, which will be reversed on June 3.
- Accounting Date: 02-Jun-2006

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Loss	105.00		105.00	
Margin Liability		105.00		105.00

Reverse the Accrual

- On June 3, the following journal entry is created to reverse the accrual from June 2.
- Accounting Date: 03-Jun-2006

Account	Entered DR	Entered CR	Accounted DR (USD)	Accounted CR (USD)
Margin Liability	105.00		105.00	
Loss		105.00		105.00



Submitting the Create Accrual Reversal Accounting Process: Explained

The Create Accrual Reversal Accounting process processes incomplete accrual reversal journal entries with accounting dates that fall on or before the End Date parameter specified. If the incomplete accrual reversal accounting entries are validated successfully, they are set to Final status and then posted to the general ledger, dependent on the parameters.

Report Parameters

Enter parameters as described in the table below.

Prompt	Description	
Subledger Application	Source system for which the Create Accrual Reversal Accounting process is being executed.	
Ledger	Ledger name for which the Create Accrual Reversal Accounting process is being executed.	
Process Category	The process category indicates that all associated accounting event classes and their accounting event types are selected for processing.	
End Date	End date puts a filter on the selection of journal entries. Only incomplete journal entries having an accounting date on or before the end date are selected for accounting.	
	Default value is current system date.	
Process Events	Adds other filter criteria for the Create Accounting process to specific events:	
	 All: Process all incomplete journal entries and meets the End Date selection criteria 	
	 Errors: Process only those entries that have previously been processed in error. 	
	 Invalid Accounts: Process only those events that have previously been processed in error. Replace any invalid accounts with the suspense account. 	
	Default value is All.	
Report	Identify the type of details to display on the execution report. The report can be printed in Summary, Detail, or No report.	
	Default value is Summary.	
Transfer to General Ledger	Indicates whether the Create Accrual Reversal Accounting process should submit the Transfer to GL process; Yes, No.	
	Default value is Yes.	
Post in General Ledger	Indicates whether to submit General Ledger posting after Transfer to GL process is complete.	
	Default value is Yes.	



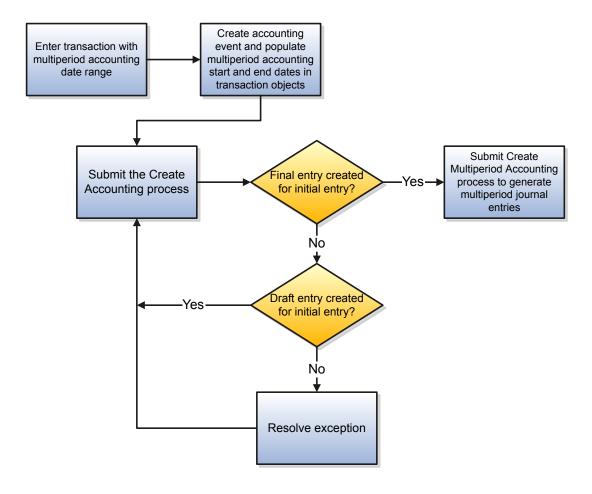
Multiperiod Accounting: Explained

Multiperiod accounting enables you to create accounting entries across more than one accounting period, for a single accounting event. The functionality is primarily used to defer the recognition of revenue or prepaid expense across multiple accounting periods.

You can:

- Determine how to distribute the amount across accounting periods.
- Specify a prepaid expense or deferral account.
- Preview multiperiod journal entries before creating and posting final entries:
 - Submit the Create Multiperiod Accounting process in Draft mode.
 - Review the multiperiod journal entries in the Create Multiperiod Accounting Execution Report.
 - Note: You can only preview multiperiod journal entries for a single accounting period.

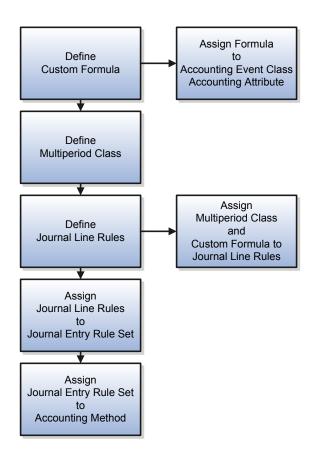
Multiperiod Accounting Process Flow:





Implementation Steps

Implementation Step Process Flow



- Define a custom formula determining how to distribute the entered amount across accounting periods. If required, you can use the new multiperiod predefined sources. For example:
 - Multiperiod Original Entered Amount
 - Multiperiod Recognized Entered Amount
 - Last Multiperiod Accounting Date
 - Number of Days in Current Accounting Period
- Define a non-multiperiod journal line rule for the deferral entry specifying the Multiperiod Class. This rule is used to
 post the fee amount to the deferral account.
 - **! Important:** First, define multiperiod classes using the Manage Subledger Accounting Lookups task.
 - Lookup Type: ORA_XLA_MULTIPERIOD_CLASS
- Define a multiperiod journal line rule for the recognition entry:
 - Enable the Multiperiod option and assign the same Multiperiod Class.



- Assign the custom formula to the Entered Amount accounting attribute.
- o Assign relevant sources to the Multiperiod Start Date and Multiperiod End Date accounting attributes.
- Assign both journal line rules to the same journal entry rule set with:
 - o A deferral account rule assigned to the non-multiperiod journal line rule.
 - o A revenue or expense account rule assigned to the multiperiod journal line rule.
- Schedule the Create Multiperiod Accounting process to run after the Create Accounting process so that recognition
 journal entries are properly booked each period.

Multiperiod Accounting: Example

The subledger multiperiod accounting and custom formula features provide the ability to satisfy the FASB 91 accounting requirement. This requirement of deferred recognition of fee and direct loan cost is met by customizing the recognition of fee amounts with custom formulas.

Example

A personal loan was originated with the following information:

• Origination date: 01-Jul-2016

Origination fee: USD 300.00

Loan duration: 6 months

Maturity date: 31-Dec-2016

Upon loan origination on 01-JUL-2016 the origination fee must be booked as unearned income:

Initial Journal Entry for Loan Fee

	Entered DR (USD)	Entered CR (USD)
Receivable	300.00	
Unearned Fee Income		300.00

The unearned fee income must be split over the life of the loan. Each portion must be recognized as income in the corresponding accounting periods, between Jul-2016 and Dec-2016. The amounts in each period would depend on the proration formula used. For example:

Monthly Journal Entry for Loan Fee

	Entered DR (USD)	Entered CR (USD)
Unearned Fee Income	50.00	
Loan Fee Income		50.00



Using Multiperiod Subledger Accounting Rules

Accounting Hub users can define subledger accounting rules to create the multiperiod accounting entries. For example:

Account Rules	Usage
Unearned Fee Income Account	Journal line rule for the deferral entry. Used to generate a credit line to the unearned fee income account.
Loan Fee Income Account	Journal line rule for the recognition entry. Used by the Create Multiperiod Accounting process to generate future entries with prorate amounts, such as the CR 50.00 (USD) in the example above.
	Unearned Fee Income Account

- Define a custom formula to calculate the total entered amount to be recognized from the multiperiod start date until the end of current accounting period. Assign the formula to the Entered Amount accounting attribute of the Loan Fee Income journal line rule.
- The Create Multiperiod Accounting process calculates entered amount for the multiperiod entries as:

Entered Amount = Amount Returned by Custom Formula - Multiperiod Recognized Entered Amount

- o Where the:
 - Multiperiod Recognized Entered Amount is calculated and stored in the Accounting Hub data model as the total amount recognized for the transaction in prior periods.
- o The approach handles any rounding differences in prorated entered currency amounts.

Sample Custom Formulas

These sample custom formula definitions can be used to prorate the loan fee amount.

Prorate by Number of Periods

• To prorate the entered amount evenly across each period:



"Fee Amount" * NumberofGLPeriod ("Origination Date", "Last Day of Current Accounting Period") / NumberofGLPeriod ("Origination Date", "Maturity Date")

- o Where the:
 - Fee Amount, Origination Date and Maturity Date are standard application sources (Application: Loans).
 - Last Day of Current Accounting Period is a system source (Application: Subledger Accounting).
 - Number of GLPeriod is a predefined Date function.
- The entered amount for the multiperiod journal would be calculated as follows:

Accounting Date	Number of Periods Since Multiperiod Start Date	Multiperiod Recognized Entered Amount (A)	Amount Returned from Custom Formula (B)	Multiperiod Journal Entered Amount (B-A)
31-Jul-16	1	0.00	300.00*1/6= 50.00	50.00
31-Aug-16	2	0.00+50.00= 50.00	300.00*2/6= 100.00	100.00-50.00= 50.00
30-Sep-16	3	50.00+50.00= 100.00	300.00*3/6= 150.00	150.00-100.00= 50.00
31-Oct-16	4	100.00+50.00= 150.00	300.00*4/6= 200.00	200.00-150.00= 50.00
30-Nov-16	5	150.00+50.00= 200.00	300.00*5/6= 250.00	250.00-200.00= 50.00
31-Dec-16	6	200.00+50.00= 250.00	300.00*6/6= 300.00	300.00-250.00= 50.00

- Note: These columns are calculated by the Create Multiperiod Accounting process:
 - Amount Returned from Custom Formula (B)
 - Multiperiod Journal Entered Amount (B-A)

Prorate by Number of Days

• To prorate entered amount by the actual number of days in each period:

"Fee Amount" * ("Last Day of Current Accounting Period" - "Origination Date" + 1) / ("Maturity Date" - "Origination Date" +1)

• The entered amount for the multiperiod journal is calculated as follows:

Accounting Date	Number of Days Since Multiperiod Start Date	Multiperiod Recognized Entered Amount (A)	Amount Returned from Custom Formula (B)	Multiperiod Journal Entered Amount (B-A)
31-Jul-16	31	0.00	300. 00*31/184. 0050.54	50.54
31-Aug-16	62	0.00+50.54= 50.54	300. 00*62/184. 00100.09	100.09-50.54= 50.55
30-Sep-16	92	50.54+50.55= 100.09	300. 00*92/184. 00150	150.00-100.09= 48.91



Accounting Date	Number of Days Since Multiperiod Start Date	Multiperiod Recognized Entered Amount (A)	Amount Returned from Custom Formula (B)	Multiperiod Journal Entered Amount (B-A)
31-Oct-16	123	100.09+48.91= 150.00	300. 00*123/184. 00200.54	200.54-150.00= 50.54
30-Nov-16	153	150.00+50.54= 200.54	300. 00*153/184. 00249.46	249.46-200.54= 48.92
31-Dec-16	184	200.54+48.92= 249.46	300. 00*184/184. 00300.00	300.00-249.46= 50.54

Note that rounding differences, if any, are included in the resulting entered amount for the multiperiod journals.

Prerequisites

- Provide date format sources to store the multiperiod start and end dates in transaction objects.
- For the event class that supports multiperiod accounting, assign the above sources to the accounting attributes:
 - Multiperiod Start Date
 - Multiperiod End Date
- Define the multiperiod class in the subledger accounting lookup. The multiperiod class is used to identify the journal line for which the amount is prorated across multiple accounting periods.
- ▼ Tip: Accounted amounts for multiperiod journals are calculated by the Create Multiperiod Accounting process prorating proportionally to the entered amount. Rounding error differences are included in the prorated amount. Multiperiod transaction data must be available in transaction objects until the last multiperiod entry has been accounted in final status.

Multiperiod Accounting: Points to Consider

To achieve a successful implementation of multiperiod accounting, the following points must be taken into consideration.

Determine when to raise a new accounting event

Determine when a new accounting event must be raised throughout the life cycle of a multiperiod transaction.

- If only the multiperiod start and end dates are changed, a new accounting event is not required. As the transaction
 object is maintained to provide the new multiperiod start date, the impact of the change is reflected in the next
 multiperiod journal.
- Raise a new accounting event if the amount is changed, or a transaction is canceled after the transaction is accounted in a Final status.

If the multiperiod start date is changed after the transaction is accounted, it's not necessary to raise a new accounting event. The assumption is that all other transaction attributes remain the same. If the transaction object is maintained to provide the new multiperiod start date, the impact of the change is reflected in the next multiperiod journal. This depends on the prorate method and is also true for future journal entries.



However, if the transaction amount is changed after the transaction is accounted; the new accounting event is needed, which affects the total deferral amount.

Vip: We suggest a detailed analysis be performed to determine whether to raise a new accounting event when transaction attributes are updated.

Raise a transaction reversal event

Consider the multiperiod journal accounting date when raising the transaction reversal event.

Set the reversal accounting date to a date after the last multiperiod journal with Final status. A reversal entry reverses all existing final entries, including the multiperiod journals, with the reversal accounting date.

For example, the transaction accounting date is on 01-Feb and multiperiod journals have been created for February and March with a Final status. If the transaction is canceled, the cancellation event should be set to 01-Apr or the first day of the next open period, whichever is later.

Provide the multiperiod start and end dates

The multiperiod start date and multiperiod end date must be provided in transaction objects for event classes that support multiperiod accounting. These date sources are mapped to the accounting attributes Multiperiod Start Date and Multiperiod End Date for the event class. They must both be blank or both be populated in the transaction objects.

Once a multiperiod journal is created in Final status, the multiperiod end date should not be modified to any day prior to the accounting date of the last multiperiod journal in the primary and secondary ledgers.

Provide the multiperiod transaction data

Multiperiod transaction data must be available in subledger transaction objects until the last multiperiod entry is accounted in Final status.

At least one line level transaction object is required for the event class to use multiperiod feature. Ensure transaction objects are populated with current data for the multiperiod transaction throughout the life cycle of multiperiod, even if the transaction is accounted.

For example, if the multiperiod start date is changed after the transaction is accounted but not all future entries have been accounted. The new multiperiod start date should be populated in the transaction object. This enables the Create Multiperiod Accounting process to calculate the prorate amount according to the modified multiperiod date range.

Process the multiperiod accounting

Multiperiod recognition entries are created based on transaction objects data and accounting rule definition at the time the Create Multiperiod Accounting process is executed.

- The entered amount value for the multiperiod journal is calculated by a formula. This formula is assigned to the Entered Amount accounting attribute on the multiperiod journal line rule.
- The accounted amount value for multiperiod journals is calculated by the Create Multiperiod Accounting process, prorating proportionally to the entered amount, including rounding differences.

Multiperiod accounting entries aren't created:

- If any subledger level reporting currency or secondary ledger is disabled, although the multiperiod transaction may not be fully recognized yet.
- If subledger level reporting currency is added after the journal is created in Final status, the multiperiod journal isn't created for the new reporting currency.



• If the transaction accounting date is prior to the first open period of a reporting currency or secondary ledger, the journal isn't created for the reporting currency or secondary ledger.

Multiperiod transactions are processed by the Create Multiperiod Accounting process only once for each accounting period and transaction.

- Don't modify the multiperiod start date for a transaction to a date in a prior accounting period of the last multiperiod journal in primary and secondary ledgers.
- If the multiperiod start date for a transaction is modified to a date in a prior accounting period, accounting periods
 that were previously processed for the transaction by the Create Multiperiod Accounting process are not processed,
 regardless of the period status. Any adjustment in the prorated amount due to change of start date will be
 accounted in the next accounting period.

Sequencing of Accounting Entries: Overview

The following sequences are attached to subledger journal entries or general ledger journal entries. These two sequences are not mutually exclusive and can coexist in the same journal entry.

Accounting Sequence

The accounting sequence is assigned to subledger journal entries at the time that the journal entry is completed. The sequence is assigned in the completion date order using the accounting date as the date criterion for determining the sequence to be used.

Reporting Sequence

The reporting sequence is assigned to both subledger journal entries and general ledger journal entries when the accounting period is closed. This sequence is used by most of the legal reports required in some countries as the main sorting criterion to display the journal entries.

Note: In some related documents, this sequence is referred to as the chronological sequence.

Related Topics

- Document Sequences: Explained
- Document Sequences: Points to Consider
- Accounting Attribute Assignments: Points to Consider
- Document Sequencing in Payables: Explained

Subledger Accounting Profile Options: Points to Consider

Set values for each profile option to specify how Oracle Fusion Subledger Accounting controls access to and processes data.



Profile Options

The following table describes the controls available for subledger accounting:

Profile Option	Profile Display Name	Profile Description
XLA_ DIAGNOSTIC_ MODE	Diagnostics Enabled	Control whether transaction data used in accounting generation should be gathered for diagnostics.
XLA_ DISABLE_ GLLEZL	Journal Import Disabled	Control whether subledger journal entries are imported to the general ledger.
XLA_ ENABLE_ TRANSFER_ TO_EBS_GL	Enable Transfer to Oracle EBS GL	Specify whether the target for transfer and post subledger journal entries is Oracle E-Business Suite General Ledger.
XLA_ ENABLE_ TRANSFER_ TO_PSFT_GL	Enable Transfer to Oracle PeopleSoft GL	Specify whether the target for transfer and post subledger journal entries is Oracle PeopleSoft General Ledger.
XLA_ OTE_ OLA_ POLLING_ INTERVAL	Online Accounting Request Polling Interval	Set the interval in seconds for online accounting engine to check for incoming accounting requests.
XLA_ OTE_OLA_PROCS	Number of Online Accounting Processes	Set the number of processes for online accounting.
XLA_ OTE_ OLA_ TIMEOUT_LIMIT	Online Accounting Processing Timeout Limit	Set the number of seconds online accounting engine attempts to process a transaction before timing out.
XLA_ SHOW_ ZERO_AMT_JRNL	Zero Amount Journal Lines Displayed	Show zero amount journal lines.

Related Topics

• Profile Options: Overview

Manage Subledger Accounting Reporting

Oracle Fusion Subledger Accounting Predefined Reports

Oracle Subledger Accounting provides accounting reports for fiscal and internal control purposes.

The reports are comprehensive from a financial standpoint and include the best source of information for each type of journal entry. These reports can be used in lieu of the General Ledger based journals and account analysis reports to see detailed subledger journal entries as well as supporting transaction information.



Scheduled Processes work area

Reports can be scheduled and run from the Scheduled Processes work area found under Tools on the Navigator.

Reports and Analytics work area

In some cases, reports are also:

- Accessed from the Reports and Analytics work area found under Tools on the Navigator or from other work areas.
- Opened using links that launch the business intelligence (BI) catalog.

The following table contains the primary predefined reports by type:

Account Analysis Reports

Display Name	Description
Account Analysis Report	Provides drill down information about the movement on a particular account for a period or range of periods. It only includes journal entries posted to general ledger.
Account Analysis by Legal Entity Report	Prints account balances by account combination and selected segments with subledger journal details, including third-party information and sequences. Flexible grouping and sorting options are provided at submission.

Journal Reports

Display Name	Description
Daily Journals Report	Lists subledger journal activity for a given period or date range, journal source, entered currency, and journal batch. Report prints detailed subledger journal lines prior to general ledger summarization. Third party and transaction details are also listed for the journal lines.
Journal Entries Report	Displays detailed information for subledger and general ledger journal entries. Groups report by ledger, journal source, journal category, and event class at submission.
Journal Ledger Report	Lists the accounting entries with subledger details like transaction number, transaction date, and line description, using flexible sorting options that are provided at report submission. This report provides a real audit trail between general ledger and subledgers to satisfy legal and business requirements.
Journals and Third Party Report	Prints all journals posted in Oracle Fusion General Ledger for the accounting period. The report provides accounting and reporting sequence details, detailed subledger accounting entry lines reporting level regardless of the ledger journals summarization. Assists in verifying that all journals are accurately recorded in chronological order with no gaps using legal sequencing rules for both accounting entries and source documents.
Subledger Detail Journal Report	Displays information about the posted general ledger journal batches that originate from Oracle Fusion Receivables and Oracle Fusion Payables subledgers. The report prints subledger details like transaction number, transaction date, transaction amount in entered and ledger currency.

Third Party Reports



Display Name	Description
Third-Party Balances Report	Displays subledger balance and account activity information for suppliers and customers.
Third-Party Account Balance Report	Prints account and original third-party transactions impacting the account during a particular period of time. The report is run to ensure that subledger and general ledger balances reconcile, and to identify possible reasons for any discrepancies.
Third-Party Balances Summary Report	Displays information for each account of the third party and third-party site, and account identification. This report can be used as a tool for auditing third-party accounts.
Third-Party Detail and Balances Report	Provides third party account balances and accounting activity details for the requested accounting period range. You can review the accounting process details by third party and audit third-party accounts in detail.

Other Subledger Accounting Reports

Display Name	Description
Accounting Event Diagnostic Report	Shows the transaction data used in accounting. The data is collected when the profile option SLA: Enable Diagnostics is set and the Create Accounting process is run.
Subledger Accounting Method Setups Report	Displays the accounting setups for a subledger application and accounting method.
Subledger Period Close Exceptions Report	Lists all accounting events and journal entries that fail period close validation. Groups report by ledger, period, journal source, journal category, and event class at submission.

To run predefined reports, navigate to the Scheduled Processes work area and follow these steps:

- 1. Click the Schedule New Process button
- 2. Search on the process name.
- **3.** Enter your parameters.
- 4. Enter your process options and schedule.
- 5. Click Submit.

Related Topics

Reports and Analytics Pane: Explained

Subledger Accounting Subject Areas, Folders, and Attributes: Explained

To create real-time analyses for Oracle Fusion Subledger Accounting you should be familiar with subject areas, folders, and attributes.

Subject Areas

To create an analysis, you begin by selecting a subject area from which you select columns of information to include in the analysis. For example, to create an analysis of journal information, you begin by selecting a Subledger Accounting- Journals



Real Time subject area. Subject areas are based around a business object or fact. In this example, the subject area is based on the column in the subledger entries tables.

Subledger Accounting has 2 specific subject areas:

- Subledger Accounting- Journals Real Time
- Subledger Accounting Supporting References Real Time

Folders

Each subject area has one fact folder and a number of dimension folders. Fact folders contain attributes that can be measured, meaning that they are numeric values like journal debit and credit amounts. Fact folders are usually at the bottom of the list of folders and are usually named after the subject area. Dimension folders contain attribute and hierarchical columns like journal name and accounting period.

Some folders appear in more than one subject area, such as Time. These are referred to as common folders or common dimensions.

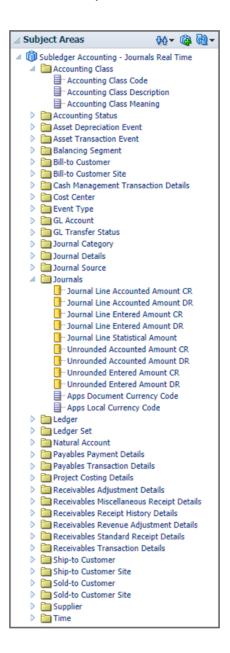
Each folder within a subject area may have a different level of granularity. For example:

- Accounting Class has accounting class attributes.
- Bill-to Customer has subfolders and attributes under the subfolders.



Attributes

Finally, each dimension folder contains attributes (columns), such as balance type and posting date. This figure illustrates the structure of subject areas, folders, and attributes.



In the preceding figure, the following components are shown:

- Subject area: Subledger Accounting Journals Real Time
- Dimension Presentation Folder: Accounting Class
- Dimension Attributes: Accounting Class Code, Accounting Class Description and Accounting Class Meaning.
- Fact Presentation Folder: Journals



• Fact - Measures: Journal Line Accounted Amount CR, Journal Line Accounted Amount DR, Journal Line Entered Amount CR, Journal Line Entered Amount DR, Journal Line Statistical Amount, Unrounded Accounted Amount CR, Unrounded Accounted Amount DR, Unrounded Entered Amount CR, and Unrounded Entered Amount DR.

Related Topics

• Data Structure for Analytics: Explained

Accessing the Create Multiperiod Accounting Execution Report: Explained

When you submit the Create Multiperiod Accounting process, the Create Multiperiod Accounting Execution Report is submitted automatically if the Create Multiperiod Accounting process completes with a success or warning status. The report displays entries created by the Create Multiperiod Accounting process.

Create Multiperiod Accounting Report

The Create Multiperiod Accounting Execution Report can be recreated.

This report can be recreated as needed using the following criteria:

- Submit the Create Multiperiod Accounting process.
- Use the same request identifier of the wanted Create Multiperiod Accounting process run previously by the same user, provided that no other Create Multiperiod Accounting process was run with the same criteria.



9 Define and Maintain Intercompany Processing Rules

Manage Intercompany System Options

Intercompany System Options: Explained

Define intercompany system options to set up intercompany processing rules at the enterprise level, based on your specific business needs.

To maintain consistency throughout an enterprise, intercompany transaction processing rules should be defined at the enterprise level. By standardizing these rules, an enterprise can minimize disputes, decrease processing time, and cut administrative costs.

Before setting up intercompany system options, you must determine how to process your intercompany transactions. For example, to:

- Enforce an enterprise-wide currency or allow intercompany transactions in local currencies.
- Allow receivers to reject intercompany transactions.
- Determine the minimum transaction amount that is processed.

System Options

Changing and saving a system option has no effect on intercompany transactions already in progress. Intercompany system options only affect new intercompany transactions, and are not retroactive to any transactions previously entered.

The system options are:

Intercompany Batch Numbering

The intercompany batch numbering option defines whether to use system generated or manual transaction batch numbering. Choose one of the following options to create intercompany batch numbers:

- System Generated: Generates intercompany batch numbers.
- o Manual: Lets you manually enter the batch number.

Minimum Transaction Amount

The minimum transaction amount represents a minimum threshold intercompany transaction amount. This amount prevents the submission of immaterial transactions for small amounts that do not add value. To implement this rule, you must first select an Intercompany currency. That currency is defaulted into the minimum transaction currency for processing intercompany transactions. These two system options ensure that when comparing a transaction amount to the minimum transaction amount, the two numbers are entered in the same currency, allowing for an accurate comparison.

Intercompany Currency

Standardize transaction processing by selecting an intercompany currency. Intercompany transactions created in the Intercompany module are always entered in this currency. This simplifies transaction processing, and eliminates



foreign conversion rate fluctuation risks. If you select an Intercompany currency, that currency is defaulted into the **Minimum Transaction Currency** field. It overwrites any other currency in that field and the field cannot be updated.

Conversion Rate Type

Select a conversion rate type that is used when transferring foreign currency intercompany transactions to General Ledger, Receivables and Payables.

Note: The foreign currency intercompany transactions are not stored in the ledger currency in the Intercompany module.

Allow Receiver to Reject Transaction

Use this system option to determine if receivers of intercompany transactions can reject transactions or not. For example, if your company policy requires intercompany transactions be approved, but does not allow receivers to reject the transactions, then you can set this system option to No.

Intercompany Calendar and Period Type

You can have an intercompany calendar that is separate from the general ledger calendar. This ensures that the opening and closing of intercompany periods can be controlled separately from the general ledger calendar. You can limit the creation of intercompany transactions by using the intercompany calendar to close periods by transaction type. For example, you can prevent users from creating intercompany sales transactions beyond the 20th of each month by closing the period for that transaction type on the 20th, but keep the period open for the other transaction types until the 25th of the month..

Select a calendar from the available general ledger calendars in the list of values.

To update the intercompany calendar to use a different general ledger accounting calendar, the intercompany period status for all intercompany transaction types must either be Never Opened or Closed.

The Period Type value is predefined from the selected accounting calendar and cannot be updated. It indicates the type of period defined for that calendar.

Default Transaction Type

Select a transaction type that will default when new intercompany batches are created.

Summarize Distributions for Receivables Invoicing

Specify this option while setting up the Intercompany System Options by selecting a Yes or No. When you select Yes, the provider distribution lines are summarized to generate one receivables invoice line. Intercompany passes the Transaction description to the receivables invoice line. Note that the accounting of the receivables invoice still shows each provider line separately.

When you select No for the Summarize Distribution for Receivables Invoicing option, each provider distribution line is passed to Receivables to be generated as a separate receivables invoice line. In this case, Intercompany passes each Provider Line description to the receivables invoice line.

Manage Intercompany Organizations



Managing Intercompany Organizations: Explained

The Manage Intercompany Organizations task allows you to define intercompany organizations and assign them to a legal entity. The intercompany organization can act either as a provider or a receiver in an intercompany transaction.

Optionally assign a receivables and payables business unit to the organization if you require invoice generation. Invoices are generated in Oracle Fusion Receivables and Oracle Fusion Payables for the business units specified.

You can initiate an intercompany transaction only for those organizations that you have access to. If a new organization is added after the application is configured, the intercompany accountant or administrator should ensure that access is given to the appropriate users.

An organization can be disabled if there are no open transactions, and all transactions for that organization are either in a New or Complete status.

Legal Entity

Each intercompany organization must be associated with a legal entity, but you can associate more than one organization to a legal entity.

Receivables and Payables Business Units

The available business units are those associated with the ledger to which the selected legal entity belongs. This assignment is optional, but is needed when the organization is engaged in intercompany activity requiring intercompany invoices.

If you enter the wrong receivables or payables business units, you can still correct them as long as the organization is not yet used in an intercompany transaction, regardless of the transaction status.

Related Topics

- Intercompany Organization Configuration: Example
- Intracompany Organization Configuration: Example

Define Invoicing Options

Customer and Supplier Assignment: Explained

The intercompany customer and supplier assignments are used to derive the customers and suppliers for intercompany invoicing. You can associate a legal entity with either a customer account or supplier, or both. Each legal entity can only have one customer account and one supplier assigned to it.

Customer Account

Assign a unique customer account to the legal entity of the organization that receives and approves intercompany transactions. The customer account type must be Internal and it must have an active bill-to site with the primary selected. Each customer account can have multiple sites each associated with a different receivables business unit. This allows invoicing for the customer in a variety of receivables business units.



Supplier

Assign a unique supplier to the legal entity of the organization that initiates intercompany transactions. The supplier must have an active primary pay site. Each supplier can have multiple sites, each associated to a different payables business unit.

Related Topics

- Intercompany Organization Configuration: Example
- Intracompany Organization Configuration: Example

Defining Receivables Assignments: Explained

Use Oracle Fusion Intercompany to generate invoices for intercompany transactions.

Defining Receivables Assignments

Set up your receivables assignments by mapping an intercompany transaction type and a receivables business unit to the receivables transaction type and receivables memo line. Oracle Fusion Receivables will use the receivables transaction type and the receivables memo line to process intercompany transactions transferred to the receivables application. You can configure specific receivables transaction types, and receivables memo lines, that you use for each intercompany transaction type for a receivables business unit.

Manage Intercompany Balancing Rules

Intercompany Balancing Rules: Explained

Intercompany balancing rules are used to generate the accounts needed to balance journals that are out of balance by legal entity or primary balancing segment values. These same rules are also used to generate the intercompany receivables account and intercompany payables account of transactions entered in the Intercompany module.

The intercompany balancing rules are also used to generate the intercompany receivables account for the provider side of an intercompany transaction. The balancing rules also used to generate the intercompany payables account for the receiver side of an intercompany transaction.

You specify the intercompany receivables and intercompany payables accounts you want to use as the template for building the intercompany receivables and intercompany payables accounts. The intercompany balancing feature then uses these rules to generate the accounts of the balancing lines it creates.

Defining Intercompany Balancing Rules

You can define intercompany balancing rules at the following levels:

- 1. Primary balancing segment
- Legal entity
- 3. Ledger
- 4. Chart of accounts

The rules are evaluated in the order shown above. For example, you can define a Primary Balancing Segment rule and a Legal Entity level rule. If both rules can be used to balance a particular journal, the Primary Balancing Segment rule is used, as it has a higher precedence.



You have flexibility in defining your intercompany balancing rules. You can have a simple setup in which you define one rule for your chart of accounts. This rule is used for all intercompany balancing for all ledgers that use this chart of accounts. Alternatively, you can have a more granular set of rules. For example, you can define a different rule for each legal entity and one chart of accounts rule to cover any gaps in your rule definitions. You can gain even more granularity by defining rules for specific journal and/or category combinations or intercompany transaction types.

Intercompany Balancing Rules: Examples

This topic provides examples of intercompany balancing rules and the intercompany balancing lines generated. These rules are used to generate the accounts needed to balance journals that are out of balance by legal entity or primary balancing segment values.

Simple Chart of Accounts

In this scenario you have one chart of accounts for all ledgers. The chart of accounts has an intercompany segment. You are using this intercompany segment and the company segment to identify the intercompany trading partners for each transaction. You do not have a need to track their intercompany activity at a granular level such as by journal source and journal category or by intercompany transaction type.

Setup

InFusion USA Chart of Accounts

Segment Qualifier	Primary Balancing Segment	Second Balancing Segment	Third Balancing Segment	Account	Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany
	(CO)	(CC)	(PROD)	(ACCT)	(IC)

Ledger, Legal Entity, Primary Balancing Segment Value Assignments

Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	InFusion Farms	3100, 3200, 3300, 3400, 3500
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Products (East)	5000
InFusion USA	InFusion Products (West)	6000
InFusion USA		1000, 9000

Chart of Accounts Rule



Rule Number	Chart of Accounts	AR Account	AP Account	Source	Category	Transaction Type
1	InFusion USA Chart of Accounts	1000 - 000 - 0000 - 13010 - 0000	1000 - 000 - 0000 - 21010 - 0000	Other	Other	None

- Journal Balancing
 - Journal before Balancing

Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
2	Liability	InFusion Textiles	4000	500	1300	40118	0000		150

- Journal Balancing
 - Journal after Balancing

Uses Rule	Line	Line Type	Legal Entity	Company	Cost Center	Product	Account	Intercom	oan _' Debit	Credit
	1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
	2	Liability	InFusion Textiles	4000	500	1300	40118	0000		150
1	3	IC AP	InFusion Farms	3100	100	1200	21010	4000		150
1	4	IC AR	InFusion Textiles	4000	500	1300	13010	3100	150	

Legal Entity and Chart of Accounts Rules

In this example the legal Entity InFusion Textiles intercompany manufacturing activities are tracked separately from its non-manufacturing activities. In order to achieve this legal entity level rules are defined specifically between the legal entity InFusion Textiles and the two manufacturing legal entities, InFusion Products (East) and InFusion Products (West). A chart of accounts rule is created to cover all other intercompany activities.

Setup

• InFusion USA Chart of Accounts



Segment Qualifier	Primary Balancing Segment	Second Balancing Segment	Third Balancing Segment	Account	Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany

• Ledger, Legal Entity, Primary Balancing Segment Value Assignments

Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	InFusion Farms	3100, 3200, 3300, 3400, 3500
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Products (East)	5000
InFusion USA	InFusion Products (West)	6000
InFusion USA		1000, 9000

• Chart of Accounts Rule

Rule Number	Chart of Accounts	AR Account	AP Account	Source	Category	Transaction Type
2	InFusion USA Chart of Accounts	1000 - 000 - 0000 - 13050 - 0000	1000 - 000 - 0000 - 21050 - 0000	Other	Other	None

Legal Entity Level Rule

Rule No.	From Legal Entity	To Legal Entity	AR Account	AP Account	Source	Category	Transaction Type
3	InFusion Textiles	InFusion Products (West)	1000 - 000 - 0000 - 13020 - 0000	1000 - 000 - 0000 - 21020 - 0000	Other	Other	None
4	InFusion Textiles	InFusion Products (East)	1000 - 000 - 0000 - 13030 - 0000	1000 - 000 - 0000 - 21030 - 0000	Other	Other	None

Journal Balancing

Journal before Balancing



Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
2	Expense	InFusion Products (East)	5000	100	1200	52340	0000	200	
3	Expense	InFusion Products (West)	6000	200	1300	52345	0000	300	
4	Liability	InFusion Textiles	4000	500	1300	40118	0000		650

Journal Balancing

Journal after Balancing

Uses Rule	Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
	1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
	2	Expense	InFusionPro (East)	od 5000	100	1200	52340	0000	200	
	3	Expense	InFusionPro (West)	od 6000	200	1300	52345	0000	300	
	4	Liability	InFusion Textiles	4000	500	1300	40118	0000		650
2	5	IC AR	InFusion Textiles	4000	500	1300	13050	3100	150	
2	6	IC AP	InFusion Farms	3100	100	1200	21050	4000		150
4	7	IC AR	InFusion Textiles	4000	500	1300	13030	5000	200	
2	8	IC AP	InFusionPro	od 5000	100	1200	21050	4000		200
3	9	IC AR	InFusion Textiles	4000	500	1300	13020	6000	300	



Uses Rule	Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
2	10	IC AP	InFusionPro (West)	d 6000	200	1300	21050	4000		300

Using Chart of Accounts Rules for Intercompany Balancing: Examples

Use chart of accounts rules for intercompany balancing. You have flexibility in defining your intercompany balancing rules with the setup of a single chart of accounts rule to use for all ledgers that use this chart of accounts.

When you create a chart of accounts rule, you specify the chart of accounts, intercompany receivables, and intercompany payables accounts you want to use, as well as the source and category. It is recommended that the intercompany receivables account be an asset type account, and the intercompany payables account be a liability type account.

You can define rules that are applied to a specific source and category, such as Payables and Invoices, or a specific intercompany transaction type, such as Intercompany Sales. Alternatively, you can choose to create rules for all sources and categories by selecting the source of **Other** and the category of **Other**.

You can have a more complex structure and define multiple rules between pairs of ledgers, legal entities, or primary balancing segment values. If you choose to have rules at various levels, then intercompany balancing evaluates the rules in the following order.

- 1. Primary balancing segment rules
- 2. Legal entity level rules
- 3. Ledger level rules
- 4. Chart of accounts rules

It is therefore recommended that you set up a chart of accounts rule for every chart of accounts structure you have. This will ensure that Intercompany Balancing will always find a rule to use to generate balancing accounts.

Intercompany Balancing will then evaluate the journal source and journal category combination in determining which rule to use for balancing. The order of precedence is as follows.

- Specific journal source and journal category
- Specific journal source and journal category of Other
- Journal source of Other and specific journal category
- Journal source of Other and journal category of Other

Chart of Accounts Rule Example

In this scenario, you choose to track intercompany balancing for companies with values 3000, and 4000 to separate intercompany accounts. You will set up specific rules are set up at the primary balancing segment value level for this. A chart of accounts rule is created for all other intercompany activity.

Setup

InFusion USA Chart of Accounts



Segment Qualifier	Primary Balancing Segment	Balancing Segment 2	Segment	Segment	Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany
	(CO)	(CC)	(PROD)	(ACCT)	(IC)

Ledger, Legal Entity, and Primary Balancing Segment Value Assignments

Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	Infusion Farms	3000
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Production	5000
InFusion USA		1000, 9000

Chart of Accounts Rule

Rule No. 1

· Chart of Accounts: InFusion USA

Source: OtherCategory: Other

Transaction Type: None

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13010	0000
AP Account	1000	000	0000	21010	000

Primary Balancing Segment Rules

Rule No. 2

• From Ledger and To Ledger: InFusion USA

• From Primary Segment Value: 3000

• To Primary Segment Value: 4000

Source: OtherCategory: Other

Transaction Type: None



IC Account	CO	CC	PROD	ACCT	IC	
AR Account	1000	000	0000	13011	0000	
AP Account	1000	000	0000	21011	0000	

Rule No. 3

· From Ledger and To Ledger: InFusion USA

• From Primary Segment Value: 3000

• To Primary Segment Value: 5000

Source: OtherCategory: Other

Transaction Type: None

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13012	0000
AP Account	1000	000	0000	21012	0000

Rule No. 4

• From Ledger and To Ledger: InFusion USA

• From Primary Segment Value: 4000

To Primary Segment Value: 3000

Source: OtherCategory: Other

Transaction Type: None

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13013	0000
AP Account	1000	000	0000	21013	0000

Rule No. 5

• From Ledger and To Ledger: InFusion USA

• From Primary Segment Value: 4000

• To Primary Segment Value: 5000

Source: OtherCategory: Other

Transaction Type: None



IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13014	0000
AP Account	1000	000	0000	21014	0000

Intercompany Balancing Lines Generated for Out of Balance Journal No. 1

Source: Manual

Category: Adjustment

Line	Account	Debit	Credit	Description	Uses Rule No.
1	5000- 100- 1200- 52330- 0000	150			
2	4000- 110- 1200- 41111- 0000		150		
3	5000- 100- 0000- 21010- 4000		150	Intercompany Payables	1
4	4000- 110- 0000- 13014- 5000	150		Intercompany Receivables	5

Intercompany Balancing Lines Generated for Out of Balance Journal No. 2

Source: Manual

Category: Adjustment

Line	Account	Debit	Credit	Description	Uses Rule No.
1	3000- 100- 1200- 52330- 0000	150			
2	4000- 110- 1200- 41111- 0000		150		
3	3000- 100- 0000- 21011- 4000		150	Intercompany Payables	2
4	4000- 110- 0000- 13013- 3000	150		Intercompany Receivables	4



Manage Secondary and Clearing Company Balancing Options

Secondary and Clearing Company Balancing Options: Explained

Secondary and Clearing Company Balancing options are used to balance the second balancing segment and/or the third balancing segment, when a transaction is unbalanced by one of these segments but is already balanced by the primary balancing segment. This option is defined for a ledger but you can create rules for various Source and Category combinations.

Secondary and Clearing Company balancing options include the following settings:

- Intercompany Receivables and Intercompany Payables accounts: You can use as the template to build balancing accounts for balancing segment 2 and balancing segment 3 when the journal is already balanced by primary balancing segment.
- Summarization options: You can choose to summarize lines within a legal entity before balancing lines are generated by choosing the Summary Net option. Alternatively choose the Detail options so lines are not summarized before balancing within a legal entity. Note that summarization always applies to balancing lines generated in a cross legal entity scenario.
- Clearing company options: Oracle recommends always setting clearing company options to handle many-to-many
 journals as this avoids balancing failing during General Ledger Posting or Subledger Accounting Create Accounting
 process.

Clearing Company Options

You can choose to set clearing company options to balance a journal with different primary balancing segment values that all belong to a single legal entity. Set the following options to handle your clearing company balancing.

- Clearing Company Condition: Choose when to use a clearing company.
 - Use clearing company only for intracompany journals.
 - Use clearing company for all many-to-many journals
 - The default value for this option is to error Many-to-Many journals.
- Clearing Company Source: Choose how the clearing company value is derived for your balancing lines, from the following options:
 - Default clearing balancing segment value.
 - Manually entered clearing balancing segment value. Manually entered clearing balancing segment value. Note that if you select Manually entered clearing balancing segment value, you will need to manually enter a value in the create journals screen. This option will not work for subledger accounting entries as they do not have a field on the user interface to enter this value.
- Clearing Company Value: If you selected Default clearing balancing segment value for Source, you must select a primary balancing segment value in this field. This is the value used to balance your intracompany or many-to-many journals.



Secondary and Clearing Company Balancing Options: Examples

This topic provides examples of ledger balancing options, the setup required, and the journal before and after balancing.

Simple Ledger Balancing with no Clearing Company Options

In this scenario the enterprise has the second balancing segment and the third balancing segment enabled for its chart of accounts. The journal is balanced by primary balancing segment but is out of balance by the second balancing segment and the third balancing segment.

Setup

InFusion USA Chart of Accounts

Segment Qualifier	Primary Balancing Segment	Second Balancing Segment	Third Balancing Segment	Account	Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany
	(CO)	(CC)	(PROD)	(ACCT)	(IC)

Ledger, Legal Entity, Primary Balancing Segment Value Assignments

Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	InFusion Farms	3100, 3200, 3300, 3400, 3500
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Products (East)	5000
InFusion USA	InFusion Products (West)	6000
InFusion USA		1000, 9000

Ledger Balancing Options

Rule Number	Ledger	Source	Category	Transaction Type	AR Account	AP Account
1	InFusion USA	Other	Other	None	1000 - 000 - 0000 - 13010 - 0000	1000 - 000 - 0000 - 21010 - 0000

- Journal Balancing
 - Journal Before Balancing



Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
2	Liability	InFusion Farms	3100	500	1300	40118	0000		150

Journal Balancing

Journal after Balancing

Uses Rule	Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
	1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
	2	Liability	InFusion Farms	3100	500	1300	40118	0000		150
1	3	AP	InFusion Farms	3100	100	1200	21010	0000		150
1	4	AR	InFusion Farms	3100	500	1300	13010	0000	150	

Ledger Balancing Options with Detail Summarization and Clearing Company Options Set

In this scenario the enterprise has the second balancing segment and the third balancing segment enabled for its chart of accounts. Management has decided to use a clearing company for balancing Many-to-Many journals only. Since the primary balancing segment values in the journal are out of balance intercompany balancing is required. Additionally, since clearing company options have been specified they will be used to balance the journal. Note that if the primary balancing segment values were balanced and only the second balancing segment and the third balancing segment were out of balance, the clearing company options would not be used.

Setup

InFusion 1000, USA Chart of Accounts

Segment Qualifier	Primary Balancing Segment	Second Balancing Segment	Third Balancing Segment		Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany

Ledger, Legal Entity, Primary Balancing Segment Value Assignments



Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	InFusion Farms	3100, 3200, 3300, 3400, 3500
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Products (East)	5000
InFusion USA	InFusion Products (West)	6000
InFusion USA		1000, 9000

• Chart of Accounts Rule

Rule Number	Chart of Accounts	AR Account	AP Account	Source	Category	Transaction Type
1	InFusion USA Chart of Accounts	1000 - 000 - 0000 - 13050 - 0000	1000 - 000 - 0000 - 21050 - 0000	Other	Other	None

Ledger Balancing Options

Rule Number	Ledger	Source	Category	Transaction Type	AR Account	AP Account
2	InFusion USA	Other	Other	None	1000 - 000 - 0000 - 13010 - 0000	1000 - 000 - 0000 - 21010 - 0000

Clearing Company Options

Rule Number	Ledger	Source	Category	Transaction Type	Condition	Source	Value
2	InFusion USA	Other	Other	None	Use for many- to-many journals only	Default clearing balancing segment value	9000

- Note: The Ledger Balancing Options and Clearing Company Options appear as one line on the page.
- Journal Balancing
 - Journal before Balancing



Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
2	Expense	InFusion Farms	3100	300	1200	52340	0000	200	
3	Expense	InFusion Farms	3300	200	1300	52345	0000	300	
4	Liability	InFusion Farms	3400	500	1300	40118	0000		320
5	Liability	InFusion Farms	3500	600	1400	40112	0000		330

Journal Balancing

Journal after Balancing

Uses Rule	Line	Line Type	Legal Entity	CO	CC	PROD	ACCT	IC	Debit	Credit
	1	Expense	InFusion Farms	3100	100	1200	52330	0000	150	
	2	Expense	InFusion Farms	3100	300	1200	52340	0000	200	
	3	Expense	InFusion Farms	3300	200	1300	52345	0000	300	
	4	Liability	InFusion Farms	3400	500	1300	40118	0000		320
	5	Liability	InFusion Farms	3500	600	1400	40112	0000		330
1	6	IC AR		9000	000	0000	13050	3100	150	
1	7	IC AP		3100	100	1200	21050	9000		150
1	8	IC AR		9000	000	0000	13050	3100	200	
1	9	IC AP		3100	300	1200	21050	9000		200
1	10	IC AR		9000	000	0000	13050	3300	300	



Type

IC AP

IC AR

IC AP

IC AR

IC AP

Entity

Rule

Manage Intercompany Transactions

Intercompany Allocations: How They Are Processed

The two processes you can use for intercompany allocation generation are Generate Intercompany Allocations and Generate General Ledger Allocations. You can use both processes for single-ledger allocations or cross-ledger allocations. The Generate Intercompany Allocations process updates Oracle Fusion Intercompany tables. The Generate General Ledger Allocations process updates Oracle Fusion General Ledger tables.

Settings That Affect Intercompany Allocations

For both the Generate Intercompany Allocations and Generate General Ledger Allocations processes, set the following parameters to create intercompany allocations:

Parameter	Generate Intercompany Allocation Process	Generate General Ledger Allocations Process
Rule or Rule set	Select the rule or rule set to create allocation calculations.	Select the rule or rule set to create allocation calculations.
Intercompany Transaction Type	Select the type of transactions that are grouped together and are identified by the type of intercompany transaction.	Not applicable.
Post Allocations	Not applicable.	Select to book amounts spread from one account to another.
Process Cross-Ledger Allocations	Not applicable.	Process allocations that run across more than one ledger.
Use Intercompany Accounts	Not applicable.	Post allocation transactions to the intercompany accounts determined by the FUN application programming interface (API).
		This option appears only for cross-ledger allocations.



Parameter	Generate Intercompany Allocation Process	Generate General Ledger Allocations Process
Use Suspense Account	Not applicable.	Post allocation transactions to the suspense account, if one exists. Cross-ledger allocations are processed using Journal Import.
		This option appears only for cross-ledger allocations.

How Intercompany Allocations are Processed Using the Generate Intercompany Allocations Process

The Generate Intercompany Allocations process:

- 1. Calls the FUN API to create intercompany transactions and process the allocation lines.
- 2. Extracts the allocation lines from the General Ledger interface table and populates the Intercompany interface tables.
- 3. Removes the allocation lines from the General Ledger interface table upon successfully populating the Intercompany interface tables.
- **4.** Generates a batch by provider legal entity and groups lines by receiver legal entity so that there is one transaction per legal entity.
- 5. Uses the entered currency amounts for populating the Intercompany interface tables.

How Intercompany Allocations are Processed Using the Generate General Ledger Allocations Process

The Generate General Ledger Allocations process:

- 1. Uses Journal Import to process the intercompany allocations if you select to post allocations for single ledger journals.
- 2. Calls the FUN API to generate the intercompany accounts if the rule or rule set contains cross-ledger lines. You select to process cross-ledger allocations using the intercompany accounts.
 - Note: If you select to process cross-ledger allocations using the suspense account, Journal Import processes the allocation lines.
- **3.** Provides the intercompany receivables or intercompany payables account lines for cross-ledger allocations going to General Ledger.
- 4. Populates the General Ledger interface table with the appropriate line for each ledger of the cross-ledger allocation.

Generating Intercompany Receivables and Intercompany Payables Accounts for Manual Transactions: Examples

The receivables (AR) and payables (AP) accounts for manual intercompany transactions are generated automatically by Oracle Fusion Intercompany. Enter distributions for the transaction and intercompany generates the receivables and payables accounts, based on the intercompany balancing rules setup.

Intercompany uses the transaction type, provider legal entity, receiver legal entity, primary balancing segment value of the first provider distribution and the primary balancing segment value of the first receiver distribution to ascertain which rule to apply. Intercompany then uses the rule, and the balancing segment values of the first provider distribution to build the intercompany



receivable account combination for the provider side of the transaction. Similarly, intercompany builds the intercompany payables account for the receiver side of the transaction, based on the first receiver distribution account.

Intercompany will evaluate the rules in the following order.

- 1. Primary balancing segment rules
- 2. Legal entity level rules
- 3. Ledger level rules
- 4. Chart of accounts rules

If there is no matching rule at the lower levels, then intercompany will use the chart of accounts rule. It is therefore recommended that you set up a chart of accounts rule for every chart of accounts structure you have. This will ensure that intercompany will always find a rule to use to generate the intercompany receivables and intercompany payables accounts for transactions.

Intercompany will then evaluate the transaction type in determining which rule to use to generate the receivables or payables account. A rule with a specific transaction type takes precedence over a rule defined for the All Other transaction type.

Generating Intercompany Receivables and Intercompany Payables Accounts for Manual Transactions Example

In this scenario you choose to track your intercompany sales for the farming and textile companies separately from other intercompany activities. Separate intercompany accounts are used for these two companies. A chart of accounts rule is created for all other intercompany activity.

Setup

InFusion USA Chart of Accounts

Segment Qualifier	Primary Balancing Segment	Balancing Segment 2	Segment	Segment	Intercompany Segment
Segment Name	Company	Cost Center	Product	Account	Intercompany
	(CO)	(CC)	(PROD)	(ACCT)	(IC)

Ledger, Legal Entity, and Primary Balancing Segment Value Assignments

Ledger	Legal Entity	Primary Balancing Segment Value
InFusion USA	InFusion Farms	3100, 3200, 3300, 3400, 3500
InFusion USA	InFusion Textiles	4000
InFusion USA	InFusion Production	5000
InFusion USA		1000, 9000

Chart of Accounts Rule



Rule No. 1

· Chart of Accounts: InFusion USA

Source: NoneCategory: None

Transaction Type: All Other

IC Account	CO	CC	PROD	ACCT	IC	
AR Account	1000	000	0000	13020	0000	
AP Account	1000	000	000	21020	000	

Legal Entity Rules

Rule No. 2

From Ledger and To Ledger: InFusion USA

• From Legal Entity: InFusion Farms

To Legal Entity: InFusion Textiles

Source: NoneCategory: None

• Transaction Type: Intercompany (IC) Sales

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13011	0000
AP Account	1000	000	0000	21011	0000

Rule No. 3

• From Ledger and To Ledger: InFusion USA

From Legal Entity: InFusion Farms

• To Legal Entity: InFusion Production

Source: NoneCategory: None

• Transaction Type: Intercompany (IC) Sales

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13012	0000
AP Account	1000	000	0000	21012	0000



Rule No. 4

From Ledger and To Ledger: InFusion USA

From Legal Entity: InFusion Textiles

• To Legal Entity: InFusion Farms

Source: NoneCategory: None

• Transaction Type: Intercompany (IC) Sales

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13013	0000
AP Account	1000	000	0000	21013	0000

Rule No. 5

From Ledger and To Ledger: InFusion USA

• From Legal Entity: InFusion Textiles

• To Legal Entity: InFusion Production

Source: NoneCategory: None

Transaction Type: Intercompany (IC) Sales

IC Account	CO	CC	PROD	ACCT	IC
AR Account	1000	000	0000	13014	0000
AP Account	1000	000	0000	21014	0000

Intercompany Accounts Generated for Intercompany Debit Transactions

Provider Distribution and Intercompany Receivable Account

This table displays the Provider side of the transaction.

Transaction	Transaction Type	Provider LE	Receiver LE	Provider Distribution	Provider AR Account Generated	Uses Rule No.
1	IC Sales	InFusion Farms	InFusion Textiles	3100- 100- 1200- 52330- 0000	3100- 100- 0000- 13011- 4000	2
2	IC Adjustments	InFusion Farms	InFusion Textiles	3100- 100- 1200- 52330- 0000	3100- 100- 0000- 13020- 4000	1



Transaction	Transaction Type	Provider LE	Receiver LE	Provider Distribution	Provider AR Account Generated	Uses Rule No.
3	IC Sales	InFusion Production	InFusion Farms	5000- 120- 1300- 52345- 0000	5000- 120- 0000- 13020- 3200	1

Receiver Distribution and Intercompany Payable Account

This table displays the Receiver side of the transaction.

Transaction	Transaction Type	Receiver LE	Provider LE	Receiver Distribution	Receiver AR Account Generated	Uses
1	IC Sales	InFusion Textiles	InFusion Farms	4000- 110- 1200- 41111- 0000	4000- 110- 0000- 21013- 3100	4
2	IC Adjustments	InFusion Textiles	InFusion Farms	4000- 110- 1200- 41111- 0000	4000- 110- 0000- 21020- 3100	1
3	IC Sales	InFusion Farms	InFusion Production	3200- 130- 1200- 41112- 0000	3200- 130- 0000- 21012- 5000	3

Intercompany Accounts Generated for Intercompany Credit Transactions

Provider Distribution and Intercompany Receivable Account

Transaction	Transaction Type	Provider LE	Receiver LE	Provider Distribution	Provider AR Account Generated	Uses Rule No.
4	IC Sales	InFusion Farms	InFusion Textiles	3100- 100- 0000- 52330- 0000	3100- 100- 0000- 13011- 4000	2
5	IC Adjustments	InFusion Farms	InFusion Textiles	3100- 100- 1200- 52330- 0000	3100- 100- 0000- 13020- 4000	1
6	IC Sales	InFusion Production	InFusion Farms	5000- 120- 1300- 52345- 0000	5000- 120- 0000- 13020- 3200	1

Receiver Distribution and Intercompany Payable Account

This table displays the Receiver side of the transaction.



Transaction	Transaction Type	Receiver LE	Provider LE	Receiver Distribution	Receiver AP Account Generated	Uses Rule No.
4	IC Sales	InFusion Textiles	InFusion Farms	4000- 100- 1200- 41111- 0000	4000- 100- 0000- 21013- 3100	4
5	IC Adjustments	InFusion Textiles	InFusion Farms	4000- 100- 1200- 41111- 0000	4000- 100- 0000- 21020- 3100	1
6	IC Sales	InFusion Farms	InFusion Production	3200- 130- 1200- 41112- 0000	3200- 130- 0000- 21012- 5000	3

Cross-Ledger Allocations: How They Are Processed

Journals can be for a single ledger or multiple ledgers within a ledger set. When you create cross-ledger allocations, they must have only one debit or one credit line. The other side of the journal can have as many lines as you need. The process adds an intercompany receivables or intercompany payables line to each ledger's journal so it can be posted. The intercompany receivables and payables accounts are generated based on the intercompany balancing rules.

Settings That Affect Allocations

General Ledger Allocations

For the General Ledger Allocations process, set the following parameters to create allocation journals:

Parameter	Description
Rule or Rule set	Select the rule or rule set to create allocation lines.
Post Allocations	Select to automatically post allocation journals after they have been imported.

Intercompany Allocations

For the Generate Intercompany Allocations process, set the following parameters to create intercompany allocations:

Parameter	Description
Rule or Rule set	Select the rule or rule set to create allocation calculations.
Intercompany Transaction Type	Select the transaction type to be used to create the intercompany transactions.

How Allocations are Processed Using the Generate General Ledger Allocations Process

The Generate General Ledger Allocations process creates journals from the allocation lines generated by the rule or rule set.



Journals can be for a single ledger or multiple ledgers. If the allocation lines span multiple ledgers each journal is balanced using the intercompany balancing rules. When you create cross-ledger allocation rules each rule must only result in either one debit line or one credit line with as many lines on the other side as you need. The process then adds intercompany receivables or intercompany payables lines to cross-ledger journals so they can be imported into the relevant ledger.

How Allocations are Processed Using the Generate Intercompany Allocations Process

The Generate Intercompany Allocations process creates an intercompany batch, transactions, provider distributions and receiver distributions from the allocation lines generated by the rule or rule set. The process creates intercompany transactions in the entered currency of the allocation lines.

The Intercompany uses the primary balancing segment values, the balancing segment to legal entity assignments and the intercompany organizations set up to create transactions from your allocation.

You can create intercompany transactions from either single ledger or cross-ledger allocation lines. To successfully process cross-ledger allocations you must have either one debit line or one credit line per allocation but as many lines as required for the other side. The single debit or single credit line forms the provider side of the transaction and the lines on the other side form the receiver side of the transaction.

Cross-Ledger Allocations: Examples

You can process cross-ledger allocations by choosing to create them as general ledger journals or intercompany transactions. Choose to generate journals from an allocation rule or rule set by submitting the Generate General Ledger Allocations process. This process provides options to balance any cross-ledger journal with a receivables or payables line.

You can also choose to create intercompany transactions from an allocation rule or rule set by submitting the Generate Intercompany Allocations process. This creates intercompany transactions that optionally can be routed to Receivables for invoice generation.

The following scenario illustrates generating balancing journal entries as well as intercompany transactions for cross-ledger allocations.

Intercompany Allocation Entries

At month end the accountant allocates a portion of any centrally incurred expenses across all organization units that contribute to, or benefit from, that expenditure, based upon a calculation that represents a reasonable allocation of how that expense should be split. By doing this allocation, the Income Statement or Profit and Loss statement for each of those organization units shows a fair representation of its share of operational costs.

In many cases, allocations only take place between departments within one subsidiary, but there may be other costs that are shared between subsidiaries on a regular basis.

For example, marketing expense is incurred within a central corporate ledger, and is allocated to the United States (US), Canadian (CA), and United Kingdom (UK) organizations based on sales volume. These organizations are separate legal entities with their own separate ledgers. The US organization bears 50% of the cost and the CA and UK organizations each bear 25% of the cost.

The Marketing Costs allocation rule is set up to generate the following allocation lines.

Ledger	Account	Debit	Credit	Description
InFusion USA	3111-110-0000-41110-0000		USD 500	Allocation Line



Ledger	Account	Debit	Credit	Description
InFusion UK	3411-000-0000-52	330-0000 USD 250		Allocation Line
InFusion Canada	3511-120-0000-52	330-0000 USD 250		Allocation Line

The intercompany balancing rules are set up to use the following accounts.

Receivables Account: 3000-000-0000-13011-0000

Payables Account: 3000-000-0000-21081-0000

Generate General Ledger Allocations Using Intercompany Accounts

Submit the General Ledger Allocations process and choose your Rule or Rule Set. Select Process Cross-Ledger Allocations and Use Intercompany Accounts options to use intercompany balancing rules to generate the receivables and payables accounts required to balance cross-ledger allocation journal lines.

The following journals are created for the Marketing Costs allocation rule.

InFusion USA journal after cross-ledger balancing:

Ledger	Account	Debit	Credit	Description
InFusion USA	3111-110-0000-411	3111-110-0000-41110-0000 USD 500		Allocation Line
InFusion USA	3111-110-0000-130	11-3411 USD 250	Cross-Ledger Intercompany Allocation with Ledger InFusion UK	
InFusion USA	3111-110-0000-130	13011-3511 USD 250		Cross-Ledger Intercompany Allocation with Ledger InFusion Canada

InFusion UK journal after cross-ledger balancing:

Ledger	Account	Debit	Credit	Description
InFusion UK	3411-000-0000-52330-	-0000 USD 250		Allocation Line
InFusion UK	3411-000-0000-21081-	3411-000-0000-21081-3111		Cross-Ledger Intercompany Allocation with Ledger InFusion USA

InFusion Canada journal after cross-ledger balancing:

Ledger	Account	Debit	Credit	Description
InFusion Canada	3511-120-0000-52330-000	Allocation Line		



Ledger	Account	Debit	Credit	Description
InFusion Canada	3511-120-0000-21081-311	1	USD 250	Cross-Ledger Intercompany Allocation with Ledger InFusion USA

Generate Intercompany Allocations

Submit the Generate Intercompany Allocations process to create intercompany transactions. If you need invoices for your allocations choose an intercompany transaction type that requires invoicing so the intercompany transactions get routed to Receivables for invoice generation.

Ledger, Legal Entity, and Primary Balancing Segment assignments are set up as follows:

Ledger	Legal Entity	Primary Balancing Segment
InFusion USA	USA Corp	3111
InFusion UK	UK Corp	3411
InFusion Canada	Canada Corp	3511

Intercompany organizations are set up as follows.

Intercompany Organization	Legal Entity
USA Sales	USA Corp
UK Sales	UK Corp
Canada Sales	Canada Corp

The following intercompany transactions are created for the Marketing Costs allocation rule.

Batch 101:

Provider	Transaction Number	Distribution Number	Distribution account Debit	Credit
USA Sales	1	1	3111-110-0000-41110-	USD 250
		2	3111-110-0000-13011- USD 250	
	2	1	3111-110-0000-41110-	USD 250
		2	3111-110-0000-13011- USD 250	



Receiver	Transaction Number	Distribution Number	Distribution account Debit	Credit
UK Sales	1	1	3411-000-0000-52330-i USD 250	
		2	3411-000-0000-21081-	USD 250
Canada Sales	2	1	3511-120-0000-52330- USD 250	
		2	3511-120-0000-21081-	USD 250

FAQs for Manage Intercompany Transactions

How can I use social networking to discuss intercompany allocation adjustments with cost center owners?

Use the Social link on the Intercompany Transactions work area to invite others to a conversation to address the adjustments.

For example, the monthly intercompany allocation of administration costs changed substantially to more accurately reflect resource usage. You need the cost center owners to validate their intercompany allocation.

From the Intercompany Transactions work area:

- 1. Search for the intercompany transaction.
- Click Social to open Oracle Social Network. Click the Share button, or click Join if collaboration has already been initiated.
- 3. Create a new related conversation.
- **4.** Invite all of the cost center owners to join the conversation.
- 5. Upload the allocation spreadsheet for the cost center owners' review.

The cost center owners can post questions and because the other cost center owners are members, they can see your responses to the questions. Each cost center owner validates their intercompany allocation in the conversation itself, which creates a lasting record.

Related Topics

What does social networking have to do with my job?

Reconcile Accounts

Managing Intercompany Reconciliation

Watch: This video tutorial shows you how to reconcile your intercompany accounts and run the intercompany reconciliation process for your reports.



Intercompany Reconciliation: Explained

Intercompany reconciliation provides you with reports to assist you with reconciling your intercompany receivables and intercompany payables accounts and identifying differences.

The main goal of the reports is to make it easy for you to identify either the receiver side or provider side of a transaction that has not been posted to the intercompany receivables or intercompany payables account.

The reports show the following intercompany lines:

- Intercompany receivables and intercompany payables lines generated by the intercompany balancing feature.
- Intercompany receivables and intercompany payables lines generated for the provider and receiver of each intercompany transaction.

The following are not included on the intercompany reconciliation reports:

- Ledger balancing lines generated when the primary balancing segment value is in balance but either the second balancing segment or the third balancing segment is out of balance
- Clearing company balancing lines

Reconciliation Reports

The reconciliation reports show the entered or transaction amount of the accounting entries booked to the intercompany receivables and payables accounts for a pair of provider and receiver legal entities. The accounted amounts may be different when the conversion rates used for the intercompany receivables and intercompany payables are different. You can select to run the reports using an additional currency and conversion rate that converts all amounts into a common currency for comparison.

The intercompany reconciliation process starts with running the **Prepare Intercompany Reconciliation Reporting Information** process. Select parameters to determine what data appears on your reports. For example, select the provider legal entity and receiver legal entity.

Once the **Prepare Intercompany Reconciliation Reporting Information** process has completed successfully. Click the Refresh icon to view the **Reconciliation Period Summary** report. This report displays the intercompany receivables and intercompany payables balances in summary for a period, and any differences between them. Drill down on the links to view the balances by source and then by journal lines. You have full drill-down capabilities to the general ledger journal, subledger accounting entry, and source receivables or payables transaction.

Generate the Reconciliation Report

To generate the Reconciliation Report, follow these steps:

- 1. Navigate to Intercompany > Reconciliation.
- 2. Click Actions > Run.
- 3. Select the report parameters.
- 4. Click **Submit** and note the process ID.
- 5. Click the Refresh icon until the report is displayed.
- **6.** The **Period Summary** report is displayed.
- Click one of the Entered Amount Difference in Transaction Currency value to view the Summary by Source report.
- 8. Review the Intercompany Receivables and Payables balances.



- 9. You can further drill down to any line by clicking the line. This displays the Journal Lines report.
- 10. Click the Journal Name to review the journal details.

Prepare Intercompany Reconciliation Reporting Information

This process extracts data used to generate reports that can be viewed and utilized to assist with reconciliation.

Prepare Intercompany Reconciliation Reporting Information Parameters

Provider Ledger

Ledger associated with the provider organization. Exclude secondary and reporting currency ledgers.

Provider Legal Entity

Legal entity of the provider organization.

Provider Accounting Period

Accounting period of the provider ledger.

Receiver Ledger

Ledger associated with the receiver organization.

Receiver Legal Entity

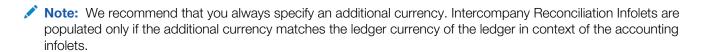
Legal entity of the receiver organization.

Receiver Accounting Period

Accounting period of the receiver ledger.

Additional Currency

Currency for converting the accounted amount.



Conversion Rate Type

Conversion rate type for the additional currency.

Conversion Date

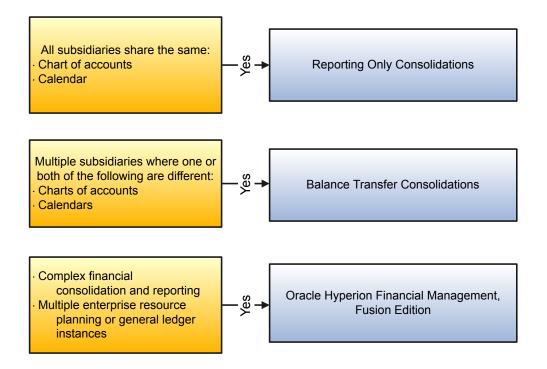
Conversion rate date for the additional currency.



10 Define Consolidations

Consolidation Method: Overview

Select the best Oracle Fusion Consolidation solution for your enterprise:



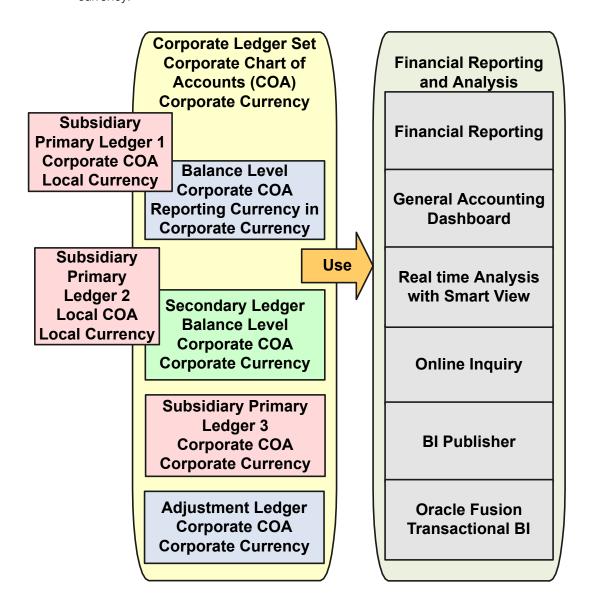
- Reporting Only Consolidations: If your subsidiaries and your corporate ledger share the same chart of accounts and calendar.
- **Balance Transfer Consolidations:** If your subsidiaries and your corporate ledger have either or both different charts of accounts and different calendars.
- Financial Management Consolidations: If there are complex factors in your financial consolidation requirements such as:
 - Complex company structures such as joint ventures, minority interest holdings, partially or fully owned subsidiaries.
 - Multiple heterogeneous systems including non general ledger data sources that are required to support nonfinancial or industry specific metrics, disclosures, and footnote schedules.



Reporting Only Consolidation Method: Explained

Use the Reporting Only Consolidation method and the reporting solutions, including Financial Reporting, Smart View, Oracle Business Intelligence (BI) Publisher, and Oracle Transactional Business Intelligence. The following scenario is illustrated in the figure.

- All subsidiaries and your corporate ledger share the same calendar.
- One of your subsidiaries has a local chart of accounts and local currency. This subsidiary uses a secondary ledger to record balances in the corporate chart of accounts and the corporate currency.
- One subsidiary has a local currency and uses reporting currency functionality to record balances in the corporate currency.





With the Reporting Only Consolidation method, perform the following tasks:

- Group the ledgers in a ledger set. This assumes the ledgers share the same chart of accounts and calendar.
- Translate balances to the corporate currency for ledgers not in the corporate currency.
 - Note: In the figure above the two subsidiary ledgers are translated to the corporate currency. The resulting reporting currency and secondary ledger are part of the ledger set for consolidation.
- Create eliminating entries.
- Report using the ledger set and the corporate currency as reporting parameters to view the consolidated balances.

If each entity's ledger has a different chart of accounts or calendar from the corporate chart of accounts and calendar, a secondary ledger is used. The secondary ledger conforms the balances to the common chart of accounts and calendar and is included in the consolidation ledger set.

Balance Transfer Consolidation Method: Explained

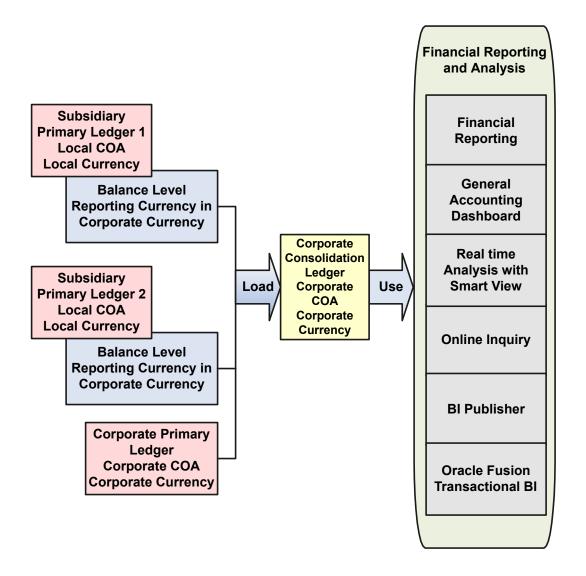
If multiple subsidiaries and the corporate ledger do not share the same chart of accounts and calendar, use the Balance Transfer Consolidation method and the reporting solutions, including Financial Reporting, Smart View, online inquiry, Oracle Business Intelligence (BI) Publisher, and Oracle Fusion Transactional Business Intelligence (Oracle Fusion Transactional BI).

The following scenario is illustrated in the figure. The subsidiaries use:

Local charts of accounts and currencies. The Corporate ledger uses a corporate chart of accounts and currency.



Balance transfers to convert the local balances to the corporate chart of accounts and currency.



With the Balances Transfer Consolidation method, perform the following tasks:

- Translate balances to the corporate currency for ledgers not in the corporate currency.
- Create a chart of accounts mapping to map subsidiaries account values to the corporate chart of accounts.
- Transfer balances from the subsidiaries to the corporate consolidation ledger using. Run the Transfer Balances
 Cross Ledgers process that transfers between any source and target ledger pair or the Balance Transfer process
 for Balance Level secondary ledgers. In the parameters, select:
 - Source and Target Ledgers
 - Chart of Accounts Mapping
 - Source Ledger and Target Ledger Period
 - Run Journal Import
 - Create Summary Journals



- Run AutoPost
- Company
- Create eliminating entries using journal entries or the Calculation Manager in the corporate consolidation ledger.
- Generate a report on the consolidated balances net of eliminations in the corporate consolidation ledger.

Reporting Only Versus Balance Transfer: Points to Consider

Here are pros and cons comparing the Reporting Only Consolidation method versus the Balance Transfer Consolidation method.

Reporting Only Consolidation Pros

:

- You are not required to run additional processes to consolidate unless ledgers have a different currency than the consolidation currency.
- View the consolidated balances anytime. This cannot be done in the Balance Transfer Consolidation method because that method requires a balance transfer be done to achieve consolidation.
- Faster close process.



Balance Transfer Consolidation Pros: Do not require a standardized chart of accounts and calendar.

Note: When reviewing balances that use either consolidation method, verify that the translation to the consolidation currency is current. If there is a journal or subledger level reporting currency ledger, translated balances are automatically available from either Reporting Only or Balance Transfer Consolidations. Only a reporting level reporting currency ledger must have the translation process run when it has a different currency than the consolidation currency.

Balance Transfer Consolidation Cons:

- May require an additional consolidation ledger to maintain balances or the current parent ledger can serve as the
 consolidation ledger. You can use your parent ledger and just transfer the subsidiary balances directly into that
 ledger.
- Must run a balance translation process if the currency is different from the consolidation currency. Then run the transfer processes to view the consolidated balances.
- Maintain charts of accounts mappings, which can be a labor intensive.
- Outdated balance transfers have to be reversed and posted, and then a new balance transfer is run every time the source ledger's balance changes.
- Requires translation to be run again if ledger currency is different than the consolidation currency.



Balance Transfers: Overview

Two methods of balance transfers are supported in Oracle Fusion General Ledger. Balance data is transferred from:

- 1. A primary ledger to a balance level secondary ledger assigned to it.
- 2. From one ledger to another without a predefined relationship.

You can drill down from the target ledger balances to the source ledger balances. The drill down can originate from:

- Financial reports.
- Smart View spreadsheet.
- Account Inspector queries.
- Account Monitor analyses.
- Journal lines in the target ledger.

When the Source and Target Ledger Currency Are the Same

You drill down on the entered amount from the **Journal Lines** page or the **Journal** page in target ledger which resulted from a balance transfer. The displayed page provides the source and target ledger details so you can analyze details. For example, analyze the accounting period and accounts used in the source ledger that transferred to the journal line amount in the target ledger.

Note: When there is a variance between the source and target ledger, there is a warning icon displayed next to the target amount and source amounts. The variance in this case could be due to journals that were posted to the source ledger after the balance transfer between source and target ledger.

When the Source and Target Ledgers Do Not Share the Same Ledger Currency

It is necessary to translate the source ledger to the target ledger's ledger currency before transferring balances. As such, balance transfers drill down also shows the reporting currency balances for the source ledger in the target ledger currency as part of the drill path.

Note: When there is a variance between the source (translated balance) and target ledger, there is a warning icon next to the target amount and source translated amounts.

The variance in this case can be due to:

- Conversion rate changes after the balance transfer.
- Journals that were posted to the source ledger after the balance transfer between source and target ledger.

Performing Consolidations: Examples

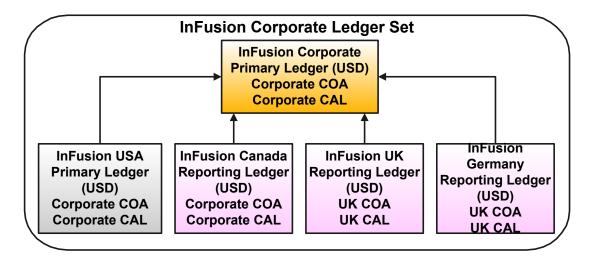
Your company, InFusion Corporation must consolidate across its entities worldwide using the Reporting Only Consolidation Method.

InFusion Corporation has four entities:

InFusion USA



- InFusion Canada
- InFusion UK
- InFusion Germany



Scenario

The four entities have different charts of accounts, calendars and currencies. InFusion Corporation uses secondary ledgers and reporting currencies to align all ledgers to the corporate chart of accounts, calendar, and currency. The InFusion Corporate ledger is an elimination ledger to hold the elimination entries. Financial Reporting functionally is used to create the consolidation reports.

Reporting Consolidation with Multiple Levels: Examples

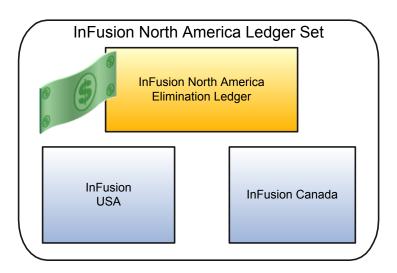
The InFusion Corporation consolidation happens at two levels.

Scenario

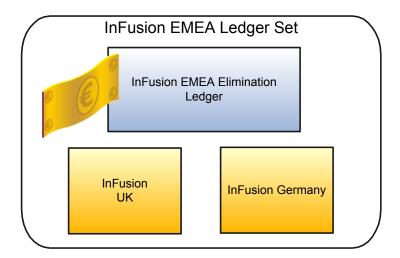
Level One



InFusion North America elimination ledger is used to record elimination entries between **InFusion USA** and **InFusion Canada**. A ledger set has been created for the three ledgers to enable creation of consolidation reports in Financial Reporting.



InFusion EMEA elimination ledger is used to record elimination entries between **InFusion UK and InFusion Germany**. A ledger set has been created for the three ledgers to enable creation of consolidation reports in Financial Reporting.

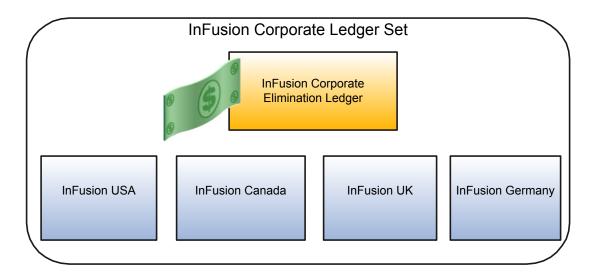


Scenario

Level Two



InFusion Corporate elimination ledger is used to record elimination entries between all four entities. A ledger set has been created for the five ledgers to enable creation of consolidation reports in Financial Reporting.



Preparing Eliminations: Examples

The following examples show how to eliminate intercompany transactions recorded in the InFusion ledgers during consolidations. The following assumptions apply to all examples.

- The arrows represent the business transactions occurring between the entities.
- The balances must be eliminated in the consolidation are between entities within a ledger set.
- The eliminations are accomplished by creating allocation rules with the Calculation Manager.
- The elimination adjustments are recorded in an elimination ledger.

Elimination Level One Example

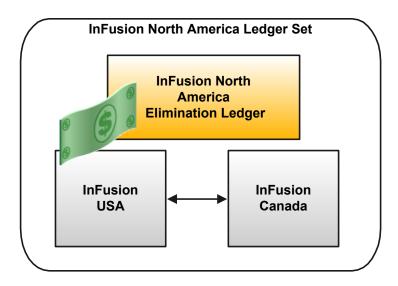
This first level of elimination entries are created for transactions between the two North America ledgers and between the two European ledgers. The elimination entries are recorded during consolidation with their respective parent ledgers.

Transaction One: InFusion USA pays InFusion Canada 10,000 USD for copper wiring.

Company	Company	Debit	Credit
InFusion USA Expense paid to InFusion Canada		10,000 USD	
	InFusion USA I/C Payable with InFusion Canada		10,000 USD



Company	Company	Debit	Credit
InFusion USA I/C Receivable with InFusion USA		10,000 USD	
	InFusion Canada Revenue received from InFusion USA		10,000 USD



InFusion North America Elimination Entry

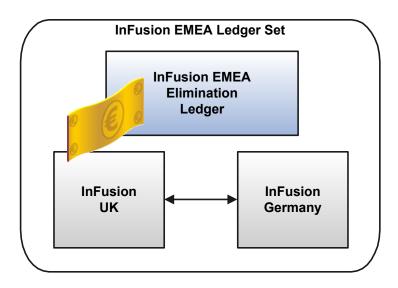
Company	Company	Debit	Credit
InFusion USA I/C Payable		10,000 USD	
	InFusion USA Expense		10,000 USD
InFusion Canada Revenue		10,000 USD	
	InFusion Canada I/C Receivable		10,000 USD

Transaction Two: InFusion UK pays InFusion Germany 5,000 EUR for manufactured technical components.

Company	Company	Debit	Credit
InFusion UK Expense paid to InFusion Germany		5,000 EUR	
	InFusion UK I/C Payable with InFusion Germany		5,000 EUR



Company	Company	Debit	Credit
InFusion Germany I/C Receivable with InFusion UK		5,000 EUR	
	InFusion Germany Revenue received from InFusion UK		5,000 EUR



InFusion EMEA Elimination Entry

Company	Company	Debit	Credit
InFusion UK I/C Payable		5,000 EUR	
	InFusion UK Expense		5,000 EUR
InFusion Germany Revenue		5,000 EUR	
	InFusion Germany I/C Receivable		5,000 EUR

Elimination Level Two Example

In addition to the two transactions above, two additional intercompany transactions took place and need to be eliminated when the four entities are all consolidated into the InFusion Corporate Elimination Ledger.

Transaction Three: InFusion Germany pays InFusion USA 20,000 USD for high technical products.



Company	Company	Debit	Credit
InFusion Germany Expense paid to InFusion USA		20,000 USD	
	InFusion Germany I/C Payable with InFusion USA		20,000 USD
InFusion USA I/C Receivable with InFusion Germany		20,000 USD	
	InFusion USA Revenue Received from InFusion Germany		20,000 USD

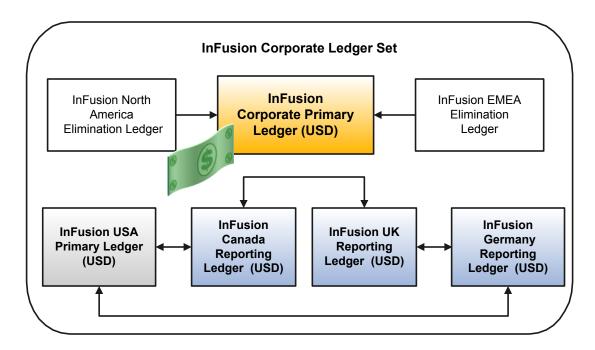
Transaction Four: InFusion Canada pays InFusion UK 15,000 USD for copper rolls.

Company	Company	Debit	Credit
InFusion Canada Expense paid to InFusion UK		15,000 USD	
	InFusion Canada I/C Payable with InFusion UK		15,000 USD
InFusion UK I/C Receivable with InFusion Canada		15,000 USD	
	InFusion UK Revenue received from InFusion Canada		15,000 USD



Final Elimination Entry at the Corporate Level

The elimination entries below are based on the previous cross ledger transactions. At different levels of the consolidation, certain intercompany payables and receivables balances need to be eliminated. Eliminations are only required in the context of a consolidation where the trading parties are both included in a given consolidation.



InFusion Corporation Elimination Entries

*(5,000 EUR 1.577 conversion rate to USD)

Company	Company	Debit	Credit
InFusion USA Payable		10,000 USD	
	InFusion Canada Receivable		10,000 USD
InFusion Germany Payable		20,000 USD	
	InFusion USA Receivable		20,000 USD
InFusion Canada Payable		15,,000 USD	
	InFusion UK Receivable		15,000 USD
InFusion UK Payable		7,885 USD*	



Company	Company	Debit	Credit
	InFusion Germany Receivable		7,885 USD*

Following is an example balance sheet showing the total elimination entries in USD.

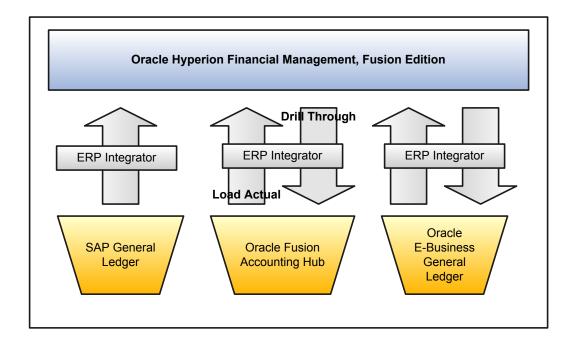
HNFUSION	

Financial Management Integration Option: Overview

The Oracle Fusion Accounting Hub includes integration to Oracle Hyperion Financial Management through the Enterprise Resource Planning (ERP) Integrator. For complex consolidation requirements: Use the integration to bring general ledger balances from the Oracle Fusion Accounting Hub to Oracle Hyperion Financial Management, Fusion Edition and perform advanced consolidation in Oracle Hyperion Financial Management.



Functionality includes drill through from Oracle Hyperion Financial Management to the Oracle Fusion Accounting Hub balances.



Perform the following tasks to implement this option:

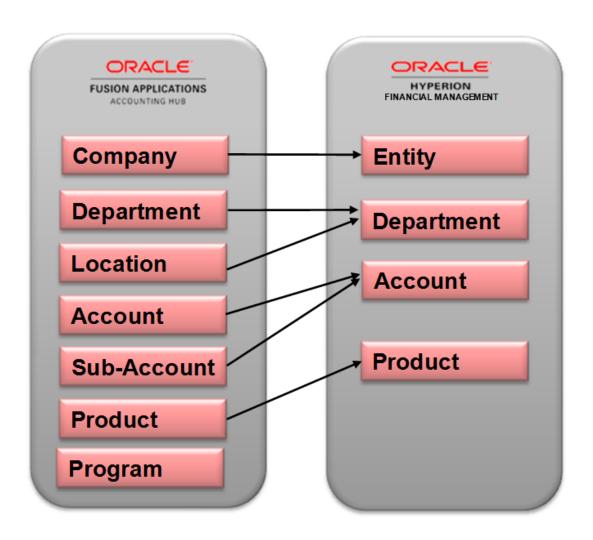
- Map chart of account values and hierarchies from the Oracle Fusion Accounting Hub to the Oracle Hyperion Financial Management, Fusion Edition dimensions.
- Load data from the general ledger balances table to Oracle Fusion Financial Management after performing the Oracle Fusion Account Hub chart of accounts to Oracle Hyperion Financial Management chart of accounts transformations.
- Perform advanced consolidation in Oracle Hyperion Financial Management.
- Drill through from Oracle Hyperion Financial Management to the Oracle Fusion Accounting Hub balances stored in the general ledger balances table.

Mapping Segments to Financial Management Dimensions: Explained

When integrating with Oracle Hyperion Financial Management, you can use the following dimensions for consolidation. Map one to one or concatenate segments into a single Oracle Hyperion Financial Management, dimensions.

Note: Data will be summarized across segments that are not mapped to Oracle Hyperion Financial Management, dimensions.





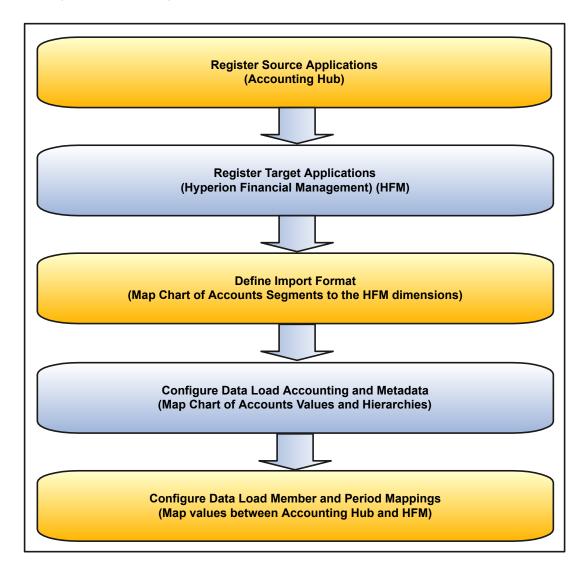
In this example:

- Company is mapped to Entity.
- Department and Location are concatenated and mapped to Department.
- Account and Sub-Account are concatenated and mapped to account.
- Product is mapped to Product.
- Program is not mapped and its data will be summarized.



Configure ERP Integrator: Overview

The following are the implementation steps that need to be performed to use the **Oracle Hyperion Financial Data Quality Management ERP Integration Adapter.**



Refer to Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide for more details on how to set up the ERP Integrator to integrate with Hyperion Financial Management.

FAQs for Define Consolidations



How can I secure balance transfer drill-down?

The balance transfer drill-down feature is secured with the same privilege that controls the existing account balance inquiry features. A specific data access set is not required to drill down from the target ledger to the source ledger to view the balance transfer information. As long as you have read or write access to the target ledger you can drill down to the source ledger. However, you are limited to just that drill path and cannot see other journals for the target ledger.



11 Define Hyperion Financial Management Integration

Oracle Hyperion Financial Management: Overview

Oracle Hyperion Financial Management is a comprehensive, web-based application that delivers global collection, financial consolidation, reporting, and analysis in one highly scalable solution. Financial Management can be licensed and integrated with the Oracle Fusion Accounting Hub to provide expanded financial consolidation, reporting, and analysis functionality. Financial Management supports these features:

- Comprehensive view of your enterprise financial information that provides key performance and operating metrics from global sources
- · Close features that include intercompany reconciliations and a consistent set of data and business measures
- Powerful multidimensional analysis that identifies and reports sources of profitability and cash flow at corporate, cost center, product, brand, customer, and channel levels
- Flexible what-if scenario management that dynamically consolidates and reports all financial budgets, forecasts and plans, producing statements as assumptions and facts change
- High-volume, preformatted reports that deliver timely including financial information for internal management and external regulatory and government bodies

The following table provides installation and configuration activities and documentation references.

Activity	Documentation
Installation of Financial Management	Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide for Oracle Hyperion Enterprise Performance Management
Configuration of Financial Management including extended analytics	 Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide for Oracle Hyperion Enterprise Performance Management Oracle Hyperion Financial Management Administrator's Guide for Oracle Hyperion Financial Management
Integration of Financial Management	Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide
License and integration with Oracle Hyperion Financial Close Management, which enables you to define, execute, and report on the interdependent activities of a financial close period. It provides centralized monitoring of all close process tasks and provides a visible, automated, repeatable process for running close processes.	Oracle Hyperion Financial Close Management Administrator's Guide for Oracle Hyperion Financial Close Management
License and integration with Oracle Hyperion Disclosure Management, which helps companies ensure the	Oracle Hyperion Disclosure Management User's Guide for Oracle Hyperion Disclosure Management



Activity

Documentation

completeness and accuracy of external and statutory financial reports, and their transformation into specifically stylized statutory filings using eXtensible Business Reporting Language (XBRL).

✓ Note: Two integration points exist between Financial Management and the Oracle Fusion Governance, Risk, and Compliance solution set. Both integrations are delivered and deployed as Oracle Fusion Governance, Risk, and Compliance Blueprint solutions. See the Oracle Fusion Governance, Risk, and Compliance documentation.

Integration with Hyperion Financial Management: Overview

For Oracle Cloud implementations, integrate with on-premise Oracle Hyperion Financial Management, Fusion Edition for advanced financial consolidations by manually exporting your balances from Oracle Fusion Accounting Hub and loading them to Oracle Hyperion Financial Management. Export your balances using either financial reports or Smart View.

For other implementations, Oracle Fusion Applications provides integration between Oracle Fusion Accounting Hub and Oracle Hyperion Financial Management through Oracle Financial Data Quality Management ERP Integrator adapter. To complete the post-installation setup for the ERP Integrator adapter, see Oracle Hyperion Financial Data Quality Management ERP Integrator Adapter for Oracle Applications Administrator's Guide.

Related Topics

- Oracle Fusion Accounting Hub: How It Works
- Oracle Fusion Accounting Hub Features: Overview



12 Define Budget Configuration

Budget Uploads: Overview

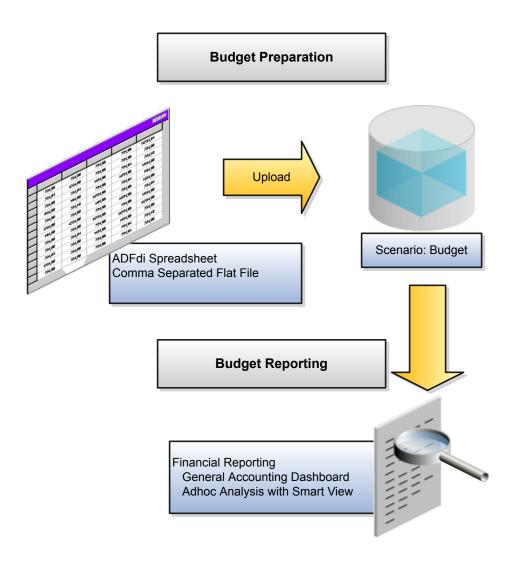
In Oracle Fusion General Ledger, you can load budget data to perform variance reporting.

If you use a third-party budgeting application or don't use a budgeting application, there are two ways to load budgets into the GL Balances Cube.

- Importing Budget Data from a Flat File: Export budget data from your budgeting application to a comma separated values .csv file. Use the Import General Ledger Budget Balances file-based data import to prepare and generate flat files in a .csv format. You can use Oracle Application Development Framework Desktop Integrator correction worksheets to correct validation errors, delete rows with errors, and resubmit the corrected error rows.
 - Note: For more information about file-based data import, see the File-Based Data Import for Oracle Financials Cloud guide.
- Importing Budget Data from a Spreadsheet: You can access the budget load spreadsheet from the General Accounting Dashboard. Enter, load, and correct budget data in the ADF Desktop Integrator spreadsheet tool. Use this tool to prepare and load budget data for multiple ledgers and periods with a common chart of accounts instance. The list of values and the web picker help you select valid values. This simplified data entry reduces errors and alerts you to errors as you enter the data in the spreadsheet. Error correction is done in the same spreadsheet.



The following figure shows the process flow from uploading a budget to reporting on it.



- ▲ Caution: When the GL Balances Cube is rebuilt, the process retains the budget balances as well as the actual balances. Only the budget balances loaded using the spreadsheet or flat file through the GL Budget Balances interface table are retained.
- Tip: Create reports in Smart View or Financial Reporting to verify that the budget data was loaded correctly.

Related Topics

• File Based Data Import for Oracle Financials Cloud



Importing Budget Data from a Flat File: Explained

Use the Upload Budgets processes to integrate budget information from other budgeting application such as Oracle Hyperion Planning, Fusion Edition. You can load your budget amounts to the General Ledger balances cube by populating the GL_BUDGET_INTERFACE table and running the Validate and Upload Budgets process. You can load budgets for multiple periods and for multiple ledgers with the same chart of accounts in a single load process. Note that the budget data is not loaded to the GL_BALANCES table and only loaded to the balances cube for variance reporting purposes.

Note: You can load data to interface tables using predefined templates and the Load Interface File for Import scheduled process, which are both part of the External Data Integration Services for Oracle Cloud. For more information about file-based data import, see the File Based Data Import guide for your cloud services.

Assigning Values for Columns in the GL_BUDGET_INTERFACE Table

You must enter values in the columns of the interface table that require values, including not null columns, for the budget import to be successful.

Enter values in the following required columns of the interface table:

Column Name	Values
RUN_NAME	Enter a name to identify the budget data set being imported.
STATUS	Enter the value NEW to indicate that you are bringing new budget data.
LEDGER_ID	Enter the appropriate ledger ID value for the budget amount. You can view the ledger ID for your ledgers in the Manage Primary Ledgers page. The ledger ID column is hidden by default, but you can display it from the View > Columns menu. If you enter multiple ledgers for the same run name, all of the ledgers must share the same chart of accounts.
BUDGET_NAME	Enter the appropriate budget name value for the budget line. You define your budget names in the Accounting Scenario value set.
PERIOD_NAME	Enter the period name that you are loading the budget data for. Note that you can load budget data to Never Opened, Future Enterable, and Open periods only.
CURRENCY_CODE	Enter the currency for your budget.
SEGMENT1 to SEGMENT30	Enter valid enabled account value for each segment in your chart of accounts.
BUDGET_AMOUNT	Enter the amount in the ledger currency for account types, expense and assets.
OBJECT_ VERSION_ NUMBER	For Oracle Cloud implementations, leave this field blank as the application automatically populates this when you load the data from the secure FTP server. For other implementations, you can set the column to a value of 1.



These columns are left as null because the budget import process either uses these columns for internal processing or does not use them currently.

- CHART_OF_ACCOUNTS_ID
- CODE COMBINATION ID
- ERROR_MESSAGE
- CREATION_DATE
- CREATED_BY
- LAST_UPDATE_DATE
- LAST UPDATE LOGIN
- LAST_UPDATED_BY
- REQUEST_ID
- LOAD REQUEST ID

Related Topics

- External Data Integration Services for Oracle Cloud: Overview
- File Based Data Import for Oracle Financials Cloud

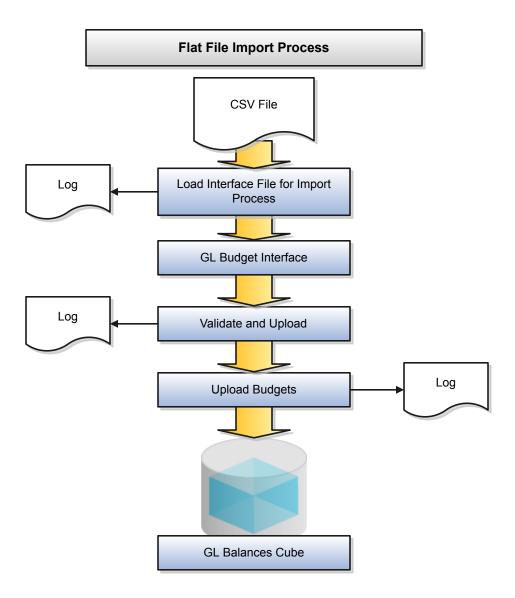
Loading Data to the Budget Interface Table: Explained

Load the budget amounts to the interface table by performing the following steps.

- 1. Export budget data from your budgeting application to a comma separated values (.csv) file. The file-based data import for Import General Ledger Budget Balances has a template that you can use. For more information about file-based data import, see the File Based Data Import for Oracle Financials Cloud guide.
- 2. Upload the zipped .csv file to the UCM directory fin/bugetBalance/import.
- 3. Launch the scheduled process called Load Interface File for Import and select the following parameters:
 - Import process: Validate and Upload Budgets
 - o Data file: Select the name of the zipped .csv file
- 4. Run the Validate and Upload Budgets process to load the budget amounts to the General Ledger balances cube.



5. Review the logs for validation errors. If there are validation errors, correct the data in the template and regenerate the .csv file. Then resubmit the data by repeating steps 3 and 4.



Related Topics

• File Based Data Import for Oracle Financials Cloud



Importing Budget Data from a Spreadsheet: Explained

Use Oracle Application Development Framework (ADF) Desktop Integrator to enter, load, and correct budget data. This functionality uses a new interface table called the GL_BUDGET_INTERFACE and requires the duty role, Budget Entry Duty.

Budget Import

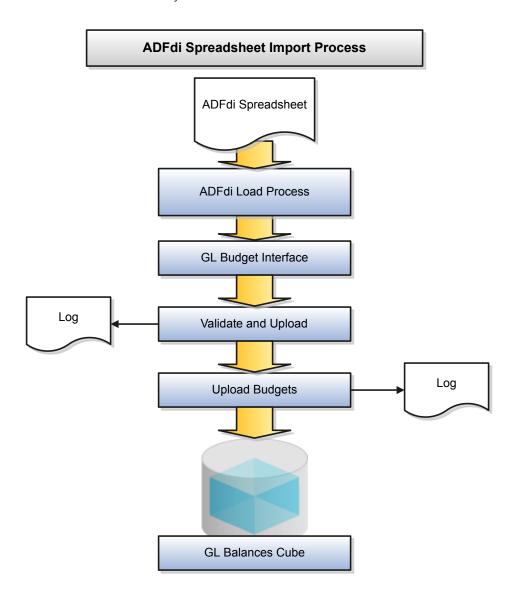
Budget Import Oracle ADF Desktop Integrator import functionality is similar to the journal import sheet in Oracle Fusion General Ledger. You may use this tool to create and upload budget data. From the General Accounting Dashboard page, download the import worksheet.

The budget import uses the Accounting Scenario value set for the budget being loaded. The Run Name is used as an identifier for the imported data set. The Oracle ADF Desktop Integrator budget import functionality:

- · Supports multiple ledgers but a single chart of accounts instance
- Allows multiple calendars and periods
- Supports entered currencies in addition to the ledger currency
- Contains user-friendly lists of values
- Performs most validations on the worksheet



Secures values by data access sets



Note: The ADF Desktop Integrator spreadsheet contains a Record Status column that shows if the rows upload successfully or with errors. Use the spreadsheet where the data was entered to enter the corrections.

Budget Correction with Oracle ADF Desktop Integrator: Explained

Oracle ADF Desktop Integrator correction functionality is similar to the journal correction sheet in Oracle Fusion General Ledger. You use this tool to correct the flat file import errors.



The correction spreadsheet functionality:

- Uses segment labels based on the data access set
- Contains user friendly lists of values
- Performs most validations on the worksheet
- Allows updating or marking the row for deletion.

Correcting Data

To use the correction spreadsheet functionality perform the following steps:

- From the General Accounting Dashboard page, you set the data access set and download the correction worksheet.
- 2. After the correction worksheet is downloaded, you query for the rows in error. Pick the run name for which there are validation errors and click on the **Search** button. This populates the budget rows in error.
- 3. Correct the rows in error or mark for deletion and submit the journal correction spreadsheet. Any errors will be reported on the worksheet.
- **4.** If the row status indicates an error, double-click it to see the error details and take necessary action. You can use the list of values to quickly correct data that is in error.

Oracle Hyperion Planning: Overview

Oracle Hyperion Planning, Fusion Edition is a comprehensive, web based, budgeting and planning application. Planning can be licensed and integrated with the Oracle Fusion Accounting Hub to provide expanded budgeting and planning functionality. Oracle Hyperion Planning, Fusion Edition:

- Delivers global collection of data and financial consolidation
- Enables entry, analysis, and reporting on data, including personalizing data entry forms
- Facilitates budgeting collaboration, communication, and control across multidivisional global enterprises
- Drives the planning process
- Provides a framework for perpetual planning, with attention to managing volatility and frequent planning cycles
- Promotes modeling using complex business rules and allocations
- Integrates with Smart View so you can design worksheets in Microsoft Excel to enter, format, analyze, and report on data

The following table provides installation and configuration activities and documentation references.

Activities	Documentation
Installation of Planning	Oracle Hyperion Enterprise Performance Management System Installation and Configuration Guide for Oracle Hyperion Enterprise Performance Management
Configuration of Planning	Oracle Hyperion Planning, Fusion Edition Administrator's Guide for Oracle Hyperion Planning
Integration of Planning	Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications Administrator's Guide



Activities

Documentation

Integration with Oracle Enterprise Planning and Budgeting Cloud Service: Overview

Integrate with Oracle Enterprise Planning and Budgeting Cloud Service to synchronize your budget and actual amounts for financial reporting and analysis.

The integration between Oracle Enterprise Planning and Budgeting Cloud Service enables you to:

- Integrate Oracle Fusion General Ledger data with the Oracle Enterprise Performance Management Cloud.
- Automate the import of data into the Oracle EPM Cloud applications.
- Schedule or manually run the integration process after selecting the source ledger from Oracle Fusion General Ledger and setting up the mappings.

For more information on completing the setup for integration with Oracle Enterprise Planning and Budgeting, Financials Cloud Service, see the Integrating Fusion Financials Cloud with EPM Cloud section in the Administering Data Management for Oracle Enterprise Performance Management Cloud guide in the Oracle Help Center (http://docs.oracle.com).





13 Define Financial Reporting

Financial Reporting Center Implementation: Points to Consider

The Financial Reporting Center adds self-service functionality for the Oracle Fusion Applications and the Oracle Fusion Accounting Hub. The financial reports are available immediately on both computers and mobile devices, which leads to quicker decision making.

Note: Store the financial reports in My Folder/ or Shared/Custom directories to access the reports in Financial Reporting Center

Job Roles and Associated Duty Roles

This table shows the predefined job roles and their associated duty roles.

Predefined Job Role	Associated Duty Role
General Accountant	Financial Reporting Management Duty RoleAccount Balances Review Duty Role
General Accounting Manager	 Financial Reporting Management Duty Role Account Balances Review Duty Role
Financial Analyst	 Financial Reporting Management Duty Role Account Balances Review Duty Role

Business Process Model Information

The features are part of the Manage Financial Reporting and Analysis detailed business process.

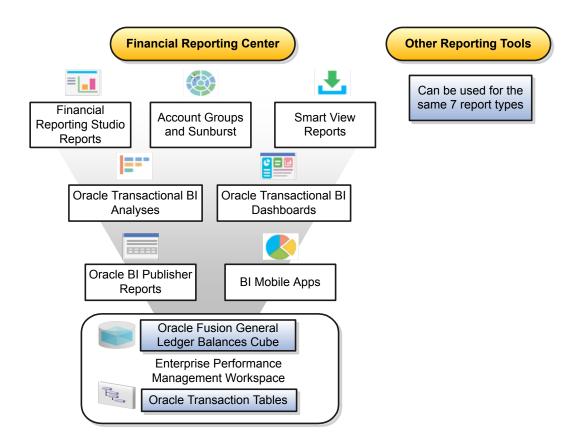
Activity	Task
Prepare Financial Reports	Generate Financial Statements
Analyze Financial Performance	Inquire and Analyze Balances

Financial Reporting Center: How It Works

The Financial Reporting Center is intended to be the primary user interface for financials end users to access all seven report types.



Financial Reporting Center Overview



Reports can be accessed through various methods. However, the Financial Reporting Center provides access to every type of report, is intended to be the primary user interface for financials end users, and is tablet and smartphone friendly.

Financial Reports are read from the **Shared** > **Custom** > **Financials** and **My Folders** directories. All other report types can be saved anywhere in the Bl Catalog however, any custom content should be in the **Shared** > **Custom** folder. Subfolders can be created within the **Shared** > **Custom** folder.

Seven types of reports can be run from the Financial Reporting Center and from the other reporting tools.

- Financial Reports: These reports are built off of the Oracle Financial Reporting Studio using data in the Oracle Fusion General Ledger balances cube. For example, company income statements and balance sheets. These reports are mainly run by users in General Ledger.
- Account Groups and Sunburst: Account groups are used to monitor key accounts in General Ledger. When a user
 creates an account group, it becomes visible in the Financial Reporting Center with the Sunburst visualization tool.
 The Sunburst visualization tool lets you interact with your account balances across various business dimensions to
 view balances from different perspectives. Account groups are used only in General Ledger.
- Smart View Reports: Smart View is a multidimensional pivot analysis tool combined with full Excel functionality. Smart View enables you to interactively analyze your balances and define reports using a familiar spreadsheet environment. These queries are mainly for users in General Ledger. To share Smart View queries, users can e-mail them to other users, or they can upload the queries to the Financial Reporting Center where users can download them to a local drive for use. The Financial Reporting Center is only a place for users to upload and download Smart View queries.



- Note: To upload a Smart View report to the Financial Reporting Center: select the Open Workspace for Financial Reports task, navigate to the Bl Catalog, and select **Upload** from the Tasks section in the left-hand pane. Be sure to upload the Excel file to one of the folder locations mentioned previously.
- Oracle Transactional Business Intelligence Analyses: These analyses and reports are built off of transactional tables using subject areas. These reports can be run by users in General Ledger, Payables, Receivables, Cash Management, Intercompany, and so on.
- Oracle Transactional Business Intelligence Dashboards: Dashboards put all the information, functions, and actions
 that a business user must have to do their job in one place. Dashboards are built off of Oracle Transactional
 Business Intelligence objects like analyses and reports. These reports can be run by users in General Ledger,
 Payables, Receivables, Cash Management, Intercompany, and so on.
- Oracle Business Intelligence Publisher Reports: Most of these reports are predefined and must first be submitted
 and resubmitted to see the latest data by the Oracle Enterprise Scheduler system through the Scheduled Processes
 navigation. These reports can be run by users in General Ledger, Payables, Receivables, Cash Management,
 Intercompany, and so on.
- BI Mobile Apps: Oracle Business Intelligence Mobile App Designer is an application that enables you to create
 multitouch information-driven applications with rich interaction, rich visualization, and rich media, for mobile devices
 such as iPhone, iPad, Android phone, tablet, and more. These reports can be run by users in General Ledger,
 Payables, Receivables, Cash Management, Intercompany, and so on.

Other Reporting Tools Overview

Six other tools are available for reporting in Financials.

The following table lists the tools and report types.

Other Reporting Tools	Report Type
General Accounting Dashboard and Account Inspector	Account Groups
Reports and Analytics	Oracle Transactional Business Intelligence Objects
Bl Catalog	All Report Types, Except Oracle Business Intelligence Publisher Reports
Enterprise Performance Management Workspace	Reports, Books, Snapshot Reports, Snapshot Books, Financial Reporting Batches, and Batch Scheduler
Enterprise Scheduler System	Oracle Business Intelligence Publisher Reports

Even though the Financial Reporting Center is designed to be the main user interface for a financial end user's reporting needs, some users may choose to use any of the six other tools for reporting in financials, such as:

- General Accounting Dashboard, which provides access to Account Groups: Uses the Account Monitor to efficiently monitor and track key account balances in real time.
- Account Inspector: Perform ad hoc queries from account groups and financial reports through drill down to underlying journals and subledger transactions.



- Reports and Analytics: This reporting tool has a panel that reflects the folder structure of the BI Catalog. Users
 can access and run any Oracle Transactional Business Intelligence analysis, report or dashboard. Users can't run
 Financial Reports or Oracle Business Intelligence Publisher reports from this interface. This interface can be used by
 all financials users.
- BI Catalog: A component of the Enterprise Performance Management Workspace where you can run all report types, except for Oracle Business Intelligence Publisher reports.
- Enterprise Performance Management Workspace: Create reports, books, snapshot reports, snapshot books, Financial Reporting batches, and batch scheduler, and schedule batches to automatically run and burst to e-mail.
- Enterprise Scheduler System: Only Oracle Business Intelligence Publisher reports can be submitted from this interface. Users access this interface by navigating to **Tools** > **Scheduled Processes**. Most financial users have access to this interface to run standard reports for General Ledger, Payables, Receivables, and so on.

Setting Up Your Financial Reporting Center: Critical Choices

Oracle Fusion Financial Reporting Center is a powerful tool for reviewing, designing, and presenting financial reports and analytic data. The critical choices required to configure and install the components in Financial Reporting Center consist of:

- Configuring the Financial Reporting Center
- Installing and configuring Financial Reporting Studio, performed by your end users
- Installing Smart View, performed by your end users
- Configuring Workspace Database Connection, performed by your administrator
- Configuring Oracle Transactional BI Dimensions

Configuring Financial Reporting Center

You have access to the reports in the Financial Reporting Center and Workspace installed with Oracle Fusion Financial Applications. Your Oracle Fusion Business Intelligence (BI) administrator defines the folder structure in Workspace. The administrator considers your company's security requirements for folders and reports, as well as report distribution requirements for financial reporting batches.

- Security can be set on folders and reports from Workspace.
- You are granted access to the folders and reports you want to view by your BI administrator.

Installing and Configuring Financial Reporting Studio

Oracle Financial Reporting Studio is client-based software. If you open Oracle Fusion Applications from Oracle Cloud, you connect to the Financial Reporting Studio through a Windows Remote Desktop Connection. Report authors download installation files for Financial Reporting Studio from Workspace by clicking Navigator > Financial Reporting Center > Tasks panel drawer > Open Workspace for Financial Reporting. Once Workspace is started, click Tools > Install > Financial Reporting Studio. After performing the prerequisites and completing the installation, start the Financial Reporting Studio. Provide your user ID, password, and the Server URL. Derive the Server URL information by following the steps:

- 1. Open Navigator > Financial Reporting Center > Tasks panel drawer > Open Workspace for Financial Reporting.
- 2. Edit the Workspace URL and remove workspace/index.jsp.



- 3. Following are two examples of **Server URLs**:
 - If the Workspace URL is https://fusionsystemtest-p-external-bi.us.oracle.com/workspace/index.jsp, the Server URL is https://fusionsystemtest-p-external-bi.us.oracle.com.
 - If the Workspace URL is https://fusionsystemtest-p-external-bi.us.oracle.com:10622/workspace/index.jsp, the Server URL is https://fusionsystemtest-p-external-bi.us.oracle.com:10622.
- 4. Copy the modified URL to the **Server URL** field.
- Note: For end users installing the Oracle Fusion Financials Reporting Studio, the installer starts a separate console window. The installer continues to run for a brief time after the installation completes the setup tasks. The process is normal, expected, and applies to Oracle Reporting Studio installations in both the Oracle Fusion Applications and Enterprise Performance Manager modes. Wait for the console window to close, which happens automatically, before clicking the **Finish** button on the Financial Reporting Studio Installation dialog box. If you click the **Finish** button before the console window closes, the Financial Reporting Studio installation may not complete.
- Note: You must save a new report before attempting to preview it with Web Preview.

Prerequisites needed for installing the Financial Reporting Studio are:

- 1. Financial Reporting Studio Client Certifications that are found at: http://www.oracle.com/technetwork/middleware/bi-foundation/hyperion-supported-platforms-085957.html.
- 2. Microsoft Office installed on your end-users computers.

Installing Smart View

Smart View is an Excel add-in that must be loaded on each client. To download Smart View, click **Navigator** > **Financial Reporting Center** > **Tasks** panel drawer > **Open Workspace for Financial Reports**. Once the workspace is started, click **Tools** > **Install** > **Smart View**.

Note: Since Smart View is an add-in to Microsoft Office products, you can install Smart View only on a Windows operating system.

Once Smart View is installed, you must configure the connection using the Smart View Shared Connections URL. You can derive the Shared Connections URL by following these steps:

- From the Financial Reporting Center task panel, select Open Workspace for Financial Reporting.
- 2. Edit the workspace URL by removing index.jsp and adding SmartViewProviders at the end.
 - Note: The following URL is an example for a Cloud-based environment. If the workspace URL is https://efops-rel5st4-cdrm-external-bi.us.oracle.com:10622/workspace/index.jsp, the Shared Connections URL is https://efops-rel5st4-cdrm-external-bi.us.oracle.com:10622/workspace/SmartViewProviders.
- 3. Copy the URL.
- 4. Open Excel.
- 5. From the Smart View menu, click **Options** > **Advanced**.
- 6. Paste the URL in the **Shared Connections URL** field.
- 7. Click OK.

For more information about configuring Smart View client for users, see the Oracle Smart View for Office User's guide.



To connect Oracle Fusion General Ledger Balances cubes in Smart View:

- Open Smart View from your Start menu > Programs > Microsoft Office > Microsoft Excel 2007.
- 2. Navigate to the **Smart View** menu > **Open**. On the **Start** on the ribbon, click on **Smart View Panel** that appears in the list of values under the ribbon. The task pane opens.
- 3. Click on the **Shared Connections** button on the task pane.
- **4.** Sign in with your user name and password.
- 5. Click on the **Select Server to proceed** list of values.
 - Note: If the Essbase Server is not there, then it has to be added. Use the following steps:
 - a. Click on the Add Essbase Server link on the bottom of the spreadsheet.
 - **b.** Specify the Essbase Server login and password.
 - **c.** Expand the Essbase server and locate the cube under it.
- **6.** Select **Oracle Essbase** from the list of shared connections.
- 7. Expand the list of cubes.
- 8. Expand your cube (name of your chart of accounts).
- 9. Click on **db**. A list of functions appears on the bottom of the panel.
- 10. Click the Ad hoc analysis function.
- Note: You must perform these steps only once for a new server and database.

To set how the name and alias of the Essbase database appears:

- 1. Click Options on the ribbon. Select Member Options > Member Name Display.
- 2. Set one of these three options:
 - o Distinct Member Name. Only shows the full Essbase distinct path.
 - Member Name and Alias: Shows both the member name and the alias.
 - Member Name Only. Shows only the member name.
- Note: The Smart Slice feature is not supported in General Ledger. For all other documentation, refer to the Oracle Smart View for Office User's Guide.

Configuring Workspace Database Connections

Administrators must create database connections from Workspace so users can access the cubes from Workspace and Financial Reporting Studio.

Note: Ledger setup has to be completed before the database connection can be created. General Ledger balances cubes are created as part of ledger setup. A separate cube is created for each combination of chart of accounts and accounting calendar. A database connection is needed for each cube.

Steps to define a database connection are:

- 1. Start at the **Navigator** by selecting **Financial Reporting Center**.
- 2. From the Financial Reporting Center task panel, select Open Workspace for Financial Reporting.
- 3. From within Workspace select the Navigator menu > Applications >BI Catalog.
- 4. Select Tools menu > Database Connection Manager.
- 5. Select **New** button.
- 6. Enter a user-friendly name for the **Database Connection Name**.



- 7. Enter Essbase as the **Type**, your server, user name, and password.
- Select Application (cube) and Database from the list of values. Expand the Application name to see the related Database, for example, db.
- 9. Click the **OK** button twice to save your selections.
- 10. Click Close button in the Database Connection Manager window to save your connection.

For more information about configuring Essbase database connections in Workspace see: Database Administrator's Guide for Oracle Essbase.

Note: The database connection is available in both Workspace and Financial Reporting Studio. Optionally, it can be set up in Financial Reporting Studio when putting grids on a report. This should only be done by an administrator.

Configuring Oracle Transactional BI Dimensions

Within Oracle Transactional Business Intelligence (BI), Accounting Segment Dimensions such as Balancing segment or Cost Center segment are based on the Chart of Accounts configuration. These segments can be configured to be tree-enabled, which means that hierarchies are defined upon the segment values for rollup purposes. In such scenarios, you must filter by a specific hierarchy when performing ad hoc queries against tree-based accounting segments. Incorrect results may occur if tree filters are not applied. To apply tree filters, create a filter condition on Tree Filter attributes in Accounting Segment Dimensions.

Note: For information, see: Administering Transactional Analyses at http://docs.oracle.com/cloud/latest/common/OATBI/toc.htm.

Migrating Financial Reports: Instructions

This process consists of two tasks:

- Exporting from the Source Instance
- Importing to the Target Instance

Exporting from the Source Instance

This process exports only the financial reports under /shared/Custom/Financials. Make sure to copy all the financial reports or the folders containing them to this folder.

To export from the source instance:

- 1. Copy all the financial reports or the folders containing them to the folder /shared/Custom/Financials.
- 2. Sign in to the Oracle Fusion Home page of the source environment with Application_Implementation_Consultant.
- 3. Select Navigator > Setup and Maintenance.
- 4. Click Manage Implementation Projects in the Tasks pane.
- 5. Click the **Create** icon to add a new implementation project.
- **6.** Enter the basic information and click **Next**.
- 7. On the Create Implementation Project: Select Offerings to Implement page, verify that the Include check box is deselected for all the projects.
- 8. Click Save and Open Project.



- 9. Click the Select and Add icon to add a task.
- 10. Select Tasks in the Search drop-down list and search on the task called Create Financial Statements.
- Select Create Financial Statements > Apply. The task is added to the Implementation Project page in the background.
- 12. Select Done.
- **13.** Click **Done** for the implementation project.
- 14. In the Tasks pane, click Manage Configuration Packages.
- **15.** Click the **Create** icon to create a new configuration project.
- **16.** Search for your implementation project in the **Name** field and enter or modify the basic information.
- 17. Select the option Setup task list and setup data.
- 18. Click Next > Submit.
- **19.** Answer Yes to the message.
- 20. Click **Refresh** until the process finishes.
- 21. Click the **Download** icon, select Download Configuration Package, and save to a local disk.

Importing to the Target Instance

This process imports the financial reports under /shared/Custom/Financials.

To import to the target instance:

- 1. Sign in to the Oracle Fusion Home page of the target environment with Application_Implementation_Consultant.
- 2. Select Navigator > Setup and Maintenance.
- 3. Click Manage Configuration Packages.
- 4. Click **Upload** to upload the configuration package that was downloaded in the export process.
- 5. Select Browse to find the file, then click Get Details, and then Submit.
- 6. In the Export and Import Processes table at the bottom of the page, click Import Setup Data to import the data.
- 7. When the Import Setup Data page appears, accept or change defaults as desired.
- 8. Click **Next** to navigate through the pages and then **Submit**.
- 9. Click **Refresh** until the process finishes.
- Note: For the Financial Reporting report definition migration service from a source to target instance, references to version IDs of dimension members hierarchies are synchronized to their version IDs in the target instance.

Financial Reporting Permissions in BI Catalog: Example

To view reporting permissions, you must have a role that inherits the **BI Administrator Role**. None of the predefined Financials job roles inherits **BI Administrator Role**.

Assign Permissions

To assign the **BI Administrator Role** and related permission use these steps:

- Select Navigator > Tools > Reports and Analytics to open the Reports and Analytics work area.
- 2. In the Contents pane, click the Browse Catalog icon. The Business Intelligence Catalog page opens.
- 3. In the Folders pane, expand Shared Folders.



- 4. Expand the **Financials** folder and then the **Bill Management** folder.
- 5. Expand Data Models. A list of reports appears on the BI Catalog.
- Under Customers Export Data Model, click More > Permissions. The Permissions dialog box opens. Scroll if
 necessary to see the complete list of permissions, which includes the role BI Administrator Role
- 7. Select a permission from the list:
 - Full Control
 - Modify
 - o Open
 - Schedule Publisher Report
 - View Publisher Output
 - No Access
 - Custom
 - Read
 - Traverse
 - Write
 - Delete
 - Change Permissions
 - Set Ownership
 - Run Publisher Report
 - Schedule Publisher Report
 - View Publisher Output
- 8. Check both:
 - Apply permissions to sub-folders.
 - Apply permissions to items within folder.
- 9. Enter Replace Options: Replace All.
- 10. Click OK.
- 11. Click the Oracle Applications: Search: Reports and Analytics tab to return to the Reports and Analytics page.

Oracle Fusion General Ledger Predefined Reports

Oracle Fusion General Ledger provides predefined reports that cover the following areas:

- Account Analysis
- Journals
- Trial Balance Reports
- Reconciliation Reports
- Chart of Accounts

You can schedule and run reports from the **Scheduled Processes** work area. In some cases, you can access and open reports in the **Reports and Analytics** work area. Both work areas are found under **Tools** on the **Navigator**. Use the icon on the top of the **Reports and Analytics** work area to open the business intelligence (BI) Catalog. You can run and edit report in the **BI Catalog**.



The following tables are the predefined reports.

Account Analysis Reports	Description
Account Analysis	 Prints balances by account segment and a secondary segment for each journal entry. Lists the subledger document number for transactions imported from subledgers.
General Ledger Account Details Report	 Provides journal information to trace each transaction back to its original source. Prints a separate page for each balancing segment value. For each journal line, prints: The account affected, the concatenated description, the journal line amount, and the beginning and ending account balance. Journal details including source, category, journal name, and effective date. Lists accounts in ascending order by account segment value. Prints a CR next to credit amounts.
Account Analysis for Contra Account Report	 Prints balances by account segment and a secondary segment. Lists the contra account for each journal entry and the subledger document number for transactions imported from subledgers. Prints by date range, accounting flexfield range, contra account, and amount range.
Average Balance Audit Account Analysis Report	 Displays the detail account activity which created the aggregate balances and related average balances. Displays daily average balance information for the selected accounts for the specified range of dates. Contains parameters such as the as-of reporting date, average balance type (period, quarter, or year average-to-date), and account ranges.

Journal Reports	Description
Journals Report	 Provides journal activity for a given period or range of periods, balancing segment value, currency, and range of account segment values. Prints the accounting date, category, journal name, reference, journal batch name, entered debit or credit amounts, net balance, and account total for each journal. Includes a total for each balancing segment and a grand total for all the activity.
General Journals Report	Provides journal activity for a given period or range of periods, balancing segment value, currency, and range of account segment values.
Journals Batch Summary Report	 Lists posted journal batches for a particular ledger, balancing segment value, currency, and date range. Provides information on actual balances for your journal batches, source, batch, and posting
	dates, total entered debits and credits. Sorts the information by journal batch within each journal entry category.
	 Includes totals each journal category and a grand total for each ledger and balancing segment value combination.
	 Does not report on budget or encumbrance balances.
Journals Details Report	Provides information on manually entered journals prior to posting, including field by field, all data entered into the applications or data imported from external sources.



Journal Reports	Description
Journals Day Book Report	 Provides posted journal entries and journal details chronologically by accounting date for a specified range of dates, journal source, and journal category. Sorts journal entries for each accounting date by document number. Prints the accounting date, document number, journal entry name, journal source and category, subledger document name and number, currency, and conversion rate. Prints for each journal line, the line number, account segment value and description, functional debit and credit amounts, description, and cost center segment value.

Trial Balance Reports	Description
Trial Balance Report	Provides summarized actual account balances and activity by ledger, balancing segment, and account segment value.
Trial Balance - Average Balances	
	 Provides a listing of ending balances and average balances for selected accounts based on an effective date specified.
	 Prints the ledger currency or foreign-entered balances.
	 Displays period, quarter, and year average-to-date balances.
	Note: Request additional information by specifying balancing segments and account ranges.

Reconciliation Reports	Description
Cash to General Ledger Reconciliation Report	Extracts cash management and general ledger accounting and transactional data for reconciling cash management to the general ledger.
Payables to Ledger Reconciliation Report	 Provides both summarized and detailed reconciling data for review. Shows payables and accounting beginning and ending balances, as well as summarized activity for the period and how this activity was accounted.
Receivables to Ledger Reconciliation Report	 Provides reconciliation of receivables data to the general ledger. Shows receivables and accounting beginning and ending balances, as well as summarized activity for the period and how the activity was accounted.

Chart of Accounts Reports	Description
Balancing Segment Value Assignments Report	 Reports on the assignment of primary balancing segment values to legal entities and ledgers across accounting set ups. Allows quick identification of overlapping balancing segment value errors and reviews of any unassigned values.
	Note: The application does not check for overlapping balancing segment values online.



Chart of Accounts Reports	Description
Chart of Accounts Mapping Rules Report	 Provides both the segment and account rules defined for a specific chart of accounts mapping. If the mapping has account rules, print each subsidiary account range and the parent account into which it maps.
	 If the mapping has segment rules, print the rule name and the parent and subsidiary segments. If the mapping has a rollup range rule, print each subsidiary segment value range and its corresponding parent segment value.

To run predefine reports, navigate to the **Scheduled Processes** work area and follow these steps:

- 1. Click the **Schedule New Process** button.
- 2. Search for your process name.
- 3. Enter your parameters.
- 4. Enter your process options and schedule.
- Click Submit.

Related Topics

• Setting Up the Reports and Analytics Pane: Procedure

General Ledger Subject Areas, Folders, and Attributes: Explained

To create real-time analyses for General Ledger, you should be familiar with subject areas, folders, and attributes.

Subject Areas

To create an analysis, you begin by selecting a subject area from which you select columns of information to include in the analysis. For example, to create an analysis of journal information, you begin by selecting a General Ledger - Journals Real Time subject area. Subject areas are based around a business object or fact. In this example, the subject area is based on the columns in the journal tables.

General Ledger has 4 general ledger specific subject areas:

- General Ledger Balances Real Time
- General Ledger Journals Real Time
- General Ledger Period Status Real Time
- General Ledger Transactional Balances Real Time
- **Tip:** You can create a report that combines data from more than one subject area. Such a report is referred to as a cross-subject area query. However, the simplest and fastest way to generate a report is to use a single subject area. If the dimension attributes and fact metrics that you are interested in are all available from a single subject area, then you can use that subject area to build the report. Such a report results in better performance and is easier to maintain. If your report requirements cannot be met by any single subject area because you need metrics from more than one subject area, you can build a cross-subject area query using common dimensions.



Folders

Each subject area has one fact folder and a number of dimension folders. Fact folders contain attributes that can be measured, meaning that they are numeric values like journal debit and credit amounts. Fact folders are usually at the bottom of the list of folders and are usually named after the subject area. Dimension folders contain attribute and hierarchical columns like journal name and accounting period.

Some folders appear in more than one subject area, such as Time. These are referred to as common folders or common dimensions.

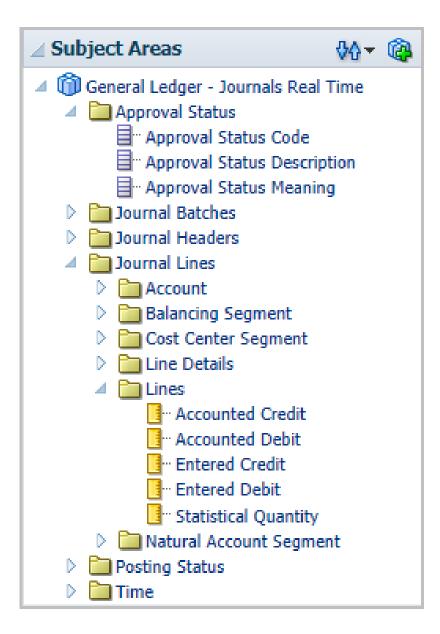
Each folder within a subject area may have a different level of granularity. For example:

- Journal Approval has approval attributes.
- Journal Batches has subfolders and attributes under the subfolders.



Attributes

Finally, each dimension folder contains attributes (columns), such as balance type and posting date. This figure illustrates the structure of subject areas, folders, and attributes.



In the preceding figure, the following components are shown:

- Subject area: General Ledger Journals Real Time
- Dimension Presentation Folder: Approval Status
- Dimension Attributes: Approval Status Code, Approval Status Description, and Approval Status Meaning.
- Fact Presentation Folder: Lines



Fact - Measures: Accounted Credit, Accounted Debit, Entered Credit, Entered Debit, and Statistical Quantity

Related Topics

Data Structure for Analytics: Explained

FAQs for Define Financial Reporting

How can I apply permissions to objects from Financial Reporting in Workspace?

Open the **Permission** dialog box from the **Tasks** list to set permissions for a catalog object. Permissions determine which user, group, or role can view, open, or modify the object. If you display this dialog box while working in the catalog in **Workspace**, any permission changes that you specify are applied immediately. If you display this dialog box as part of the **Batch Scheduler** wizard, then the permission changes are not applied until you run the batch.

Video Tutorials to Define a Basic Financial Report

Defining a Basic Financial Report

Watch: This video tutorial shows you how to create a financial report to analysis and report results of business transactions.

Adding Formulas to a Financial Report

Watch: This video tutorial shows you how to add formulas to a financial report to calculate balances from business transactions.

Defining Range Functions for a Financial Report

Watch: This video tutorial shows you how to create range functions in a financial report to span several accounting periods.

Adding Grid Points of View for a Financial Report

Watch: This video tutorial shows you how to define grid points of view in a financial report to reduce user input at report run time.



Setting the Page and Grid Properties for a Financial Report

Watch: This video tutorial shows you how to define the page, row, and column attributes in a financial report.

Adding Formatting and Graphing to a Financial Report

Watch: This video tutorial shows you how to format and add graphs to a financial report to improve analysis of the data.



14 Define Extensions for Fusion Accounting Hub

Define Custom Enterprise Scheduler Jobs for Fusion Accounting Hub

Managing Job Definitions: Highlights

Users run scheduled processes based on Oracle Enterprise Scheduler jobs to process data and, in some cases, to provide report output. A job definition contains the metadata that determines what the job does and what options are available to users. You can create and edit job definitions in the Setup and Maintenance work area, using the Manage Custom Enterprise Scheduler Jobs task for your application.

Viewing Job Definitions

- Use the Manage Job Definitions tab to access predefined and custom job definitions.
- The Name column shows an asterisk for predefined job definitions.

Creating Job Definitions

- You or a technical administrator can create jobs based on Oracle Business Intelligence Publisher reports, Java, PL/SQL, or any other supported technology.
- Every predefined or custom job must have a job definition.
- For Oracle Cloud implementations, you can create custom job definitions only for custom jobs based on reports.
- The Enable submission from Enterprise Manager check box is not applicable to Oracle Cloud implementations.
 - o If you don't select this check box, then the job can't be run from Enterprise Manager.
 - If you select this check box, then you can define parameters for your job definition only in Enterprise Manager.
 Save the rest of your work on the job definition, and then go to Enterprise Manager if you need to define parameters.

Editing Job Definitions

- You can edit all aspects of custom job definitions.
- For predefined job definitions, you can:
 - Determine if user properties are read-only or not.
 - Edit what are described as job properties in the Oracle Fusion Applications Extensibility Guide for Developers.
 - See: Customizing Existing Oracle Enterprise Scheduler Job Properties

Related Topics

Managing Job Sets: Highlights



How can I see which applications a Manage Custom Enterprise Scheduler Jobs task includes?

Managing List of Values Sources: Highlights

A list of values source for Oracle Enterprise Scheduler job definitions determines where a list of values comes from and what the specific values are. Use these lists for parameters and application defined properties, for example a list of countries that users can choose from for a Country parameter.

Note: Since you can't edit parameters for predefined job definitions, list of values sources are only for parameters in custom job definitions.

Accessing List of Values Sources

- Access list of values sources in the Setup and Maintenance work area, using the Manage Custom Enterprise Scheduler Jobs task for your application.
- Open the Manage List of Values Sources tab.

Creating and Editing List of Values Sources

- Search for list of values sources to edit or delete, or to make sure a particular source doesn't already exist before you create it.
- Create list of values sources to register them for use in job definitions.

Related Topics

- Managing Job Sets: Highlights
- How can I see which applications a Manage Custom Enterprise Scheduler Jobs task includes?



Glossary

account rule

The rule that processing uses to derive complete accounts or segment values on a subledger journal entry. Conditions can be defined within the rule to derive a different account based on specific attributes of the transaction.

accounting attribute

Predefined fields that map to components of subledger journal entries. Sources are assigned to accounting attributes.

accounting class usage

Grouping of accounting classes used in subledger journal entry reporting or processing.

accounting event class

Categories that classify transaction types and group event types for accounting rules.

accounting event type

Represents a business operation that may have an accounting impact.

accounting method

A set of journal entry rules which determine how a subledger journal entry is created for each event class or event type.

accounting period

The fiscal period used to report financial results, such as a calendar month or fiscal period.

API

Abbreviation for application programming interface.

application feature

A standardized functionality that is available to implemented.

application identity

Predefined application level user with elevated privileges. An application identity authorizes jobs and transactions for which other users are not authorized, such as a payroll run authorized to access a taxpayer ID while the user who initiated the job is not authorized to access such personally identifiable information.



AutoPost criteria sets

A grouping of options and submission frequencies used to select journal entries for automatic posting.

balances cube

A multidimensional database that holds accounting financial data. The cube allows different views of the balances to be quickly displayed.

balancing segment

A chart of accounts segment used to automatically balance all journal entries for each value of this segment.

business object

A resource in an enterprise database, such as an invoice or purchase order.

business unit

A unit of an enterprise that performs one or many business functions that can be rolled up in a management hierarchy.

chart of accounts

The account structure your organization uses to record transactions and maintain account balances.

clearing company

The intercompany clearing entity used to balance the journal.

constant

Holds the numeric value used to evaluate numeric conditions in Contract Expert rules. A constant permits you to reset the conditions of many rules with just one edit.

context

A grouping of flexfield segments to store related information.

context segment

The flexfield segment used to store the context value. Each context value can be associated with a different set of context-sensitive segments.

context-sensitive segment

A flexfield segment that may or may not appear depending upon a context. Context-sensitive segments are custom attributes that apply to certain entity rows based on the value of the context segment.



conversion rate

Ratio at which the principal unit of one currency can be converted into another currency.

corporate rate type

Rate you define to standardize rates used in conversion of one currency to another over a period of time. This rate is generally a standard market rate determined by senior financial management for use throughout the organization.

cost center

A unit of activity or a group of employees used to assign costs for accounting purposes.

country holding company

A legal entity that acts on behalf of several divisions within an enterprise, and is the legal employer in a country.

cube

A block of data that contains three or more dimensions. An Essbase database is a cube.

department

A division of a business enterprise dealing with a particular area of activity.

description rule

The rule that defines description content that can appear on the subledger journal header and line.

descriptive flexfield

Customizable expansion space, such as fields used to capture additional descriptive information or attributes about an entity, such as a customer case. You may configure information collection and storage based on the context.

dimension

See

division

A business-oriented subdivision within an enterprise. Each division is organized to deliver products and services or address different markets.



document sequence

A unique number that is automatically or manually assigned to a created and saved document.

enterprise

An organization with one or more legal entities under common control.

ESS

Acronym for Enterprise Storage Server. An application that optimizes data storage.

fixed rate type

Rate you set between two currencies that remains constant. For example, a rate set between the euro currency and each Economic and Monetary Union (EMU) currency during the conversion to the euro currency.

flexfield

A flexible data field that you can customize to contain one or more segments or store additional information. Each segment has a value and a meaning.

flexfield segment

An extensible data field that represents an attribute and captures a value corresponding to a predefined, single extension column in the database. A segment appears globally or based on a context of other captured information.

item master

A collection of data that describes items and their attributes recorded in a database file.

iob

A generic role that is independent of any single department or location. For example, the jobs Manager and Consultant can occur in many departments.

job definition

The metadata that determines what a job does and what options are available to users when they submit the scheduled process. A job is the executable for a scheduled process.

journal

An element of a journal entry consisting of the name, accounting date, category, ledger, and currency for single currency journal entries. Used to group journal lines.

journal batch

An element of a journal entry consisting of the name, source, and accounting period. Used to group journals for processing and easier querying.

journal category

A name used to group journal entries with similar characteristics, such as adjustments, accruals, or reclassifications.



journal entry

Point of entry of business transactions into the accounting system. Chronological record, with an explanation of each transaction, the accounts affected, and the amounts to increase or decrease each account.

journal line

An element of journal entries consisting of account combinations and credit or debit amounts. Optionally, contains statistical quantities, currency information for multicurrency journals, and additional information.

journal line rule

A rule that includes options to convert transactional data into a subledger journal line. Conditions can be defined within the rule so it's only used based on specific attributes of a transaction.

journal source

A name that indicates the origin of journal entries, such as payables, receivables, or manual. Used as an attribute in automatic posting and journal import processes.

key flexfield

Configurable flexfield comprising multiple parts or segments, each of which has a meaning either individually or in combination with other segments. Examples of key flexfields are part numbers, asset category, and accounts in the chart of accounts.

legal authority

A government or legal body that is charged with powers such as the power to make laws, levy and collect fees and taxes, and remit financial appropriations for a given jurisdiction.

legal employer

A legal entity that employs people.

legal entity

An entity identified and given rights and responsibilities under commercial law through the registration with country's appropriate authority.

legal jurisdiction

A physical territory, such as a group of countries, single country, state, county, parish, or city, which comes under the purview of a legal authority.

legal reporting unit

The lowest level component of a legal structure that requires registrations. Used to group workers for the purpose of tax and social insurance reporting or represent a part of your enterprise with a specific statutory or tax reporting obligation.

legislative data group

A means of partitioning payroll and related data. At least one legislative data group is required for each country where the enterprise operates. Each legislative data group is associated with one or more payroll statutory units.



line of business

Set of one or more highly related products which service a particular customer transaction or business need. Refers to an internal corporate business unit.

mainline metadata

The primary branch of metadata that a sandbox is published to. Once published, changes made in the sandbox become available to all users.

mapping set

Maps a combination of input source values to specific output values. The output value of a mapping set is used to derive accounts or segments in account rules.

natural account

Categorizes account segment values by account type, asset, liability, expense, revenue, or equity, and sets posting, budgeting, and other options.

natural account segment

A chart of accounts segment used to categorize your accounting transactions by account type: asset, liability, owner's equity, revenue, or expense.

offering

A comprehensive grouping of business functions, such as Sales or Product Management, that is delivered as a unit to support one or more business processes.

payroll statutory unit

A legal entity registered to report payroll tax and social insurance. A legal employer can also be a payroll statutory unit, but a payroll statutory unit can represent multiple legal employers.

PL/SQL

Abbreviation for procedural structured queried language.

position

A specific occurrence of one job that is fixed within one department. It is also often restricted to one location. For example, the position Finance Manager is an instance of the job Manager in the Finance Department.

post

Update account balances by the recorded debit or credit amount in each journal entry to the related accounts in the general ledger.

primary balancing segment value

A segment value used to represent a legal entity in the chart of accounts and automatically balance all intercompany and intracompany transactions and journal entries.



primary ledger

Main record-keeping ledger.

process category

Group of one or more logically related event classes that can be used to restrict which events are processed by the Create Accounting process.

reference data

Data in application tables that is not transactional or high-volume, which an enterprise can share across multiple organizations. For example, sales methods, transaction types, or payment terms.

reference data set

Contains reference data that can be shared across a number of business units or other determinant types. A set supports common administration of that reference data.

reference object

Standardized data model containing reference information owned by other subledger applications and used by the Create Accounting process to create subledger journal entries from accounting events.

registration

The record of a party's identity related details with the appropriate government or legal authorities for the purpose of claiming and ensuring legal and or commercial rights and responsibilities.

sandbox

A testing environment that isolates untested code changes from the mainline environment so that these changes don't affect the mainline metadata or other sandboxes.

scheduled process

A program that you run to process data and, in some cases, generate output as a report.

secondary ledger

An optional, additional ledger that is associated with the primary ledger for an accounting setup. Secondary ledgers can represent the primary ledger's data in another accounting representation. The Secondary ledger differs in chart of accounts, accounting calendar, currency, subledger accounting method and ledger processing options.

segment

A segment is a single field within a flexfield and maps to a single table column in your database. When customizing a flexfield, you define the appearance and meaning of individual segments.



set

Classified and grouped reference data that organizational entities share.

source

Contextual and reference information from subledger applications used in conjunction with accounting rules to create subledger journal entries.

source assignment

Assignment of sources to one or more event classes. Sources are only available for creating accounting rules for the event classes to which they are assigned.

source system

A system that a non-Oracle software provider provides, or an internal system that creates events that the Oracle Fusion Accounting Hub uses.

spot rate type

Rate you enter to perform conversion based on this rate as of a specific date. This rate applies to the immediate delivery of a currency.

subject area

A set of columns, or pieces of data, related to a specific business object or area.

subledger

A low-level ledger that stores and manages the details that substantiate the monetary value stored in the general ledger. Oracle Fusion Receivables and Oracle Fusion Payables are examples of subledgers.

subledger accounting options

Defines how certain accounting processing should be done for transactions at the ledger and subledger application level.

subledger journal entry

A detailed journal entry generated for a transaction in a subledger application.

subledger journal entry line

An individual debit or credit line that is part of a subledger journal entry.

subledger journal entry rule set

A set of rules defining how to generate a complete journal entry for an accounting event.

supporting reference

Stores additional source information about a subledger journal entry line which can be used to establish a subledger balance for source values for an account.



third party

The external party that is defined in a business relationship to provide or receive goods and services.

trading partner

An external party, such as a supplier, in the Oracle B2B application for which electronic documents are sent or from which documents are received. A trading partner in Oracle B2B corresponds to a supplier site.

transaction object

Standardized data model containing transaction information used by the Create Accounting process to create subledger journal entries from accounting events.

tree

Information or data organized into a hierarchy with one or more root nodes connected to branches of nodes. A tree must have a structure where each node corresponds to data from one or more data sources.

tree structure

A set of guidelines or a framework applied to create a tree, include data, version a tree, or access a tree.

tree version

An instance of a tree that includes life cycle elements such as start and end dates, and indicates whether the tree is active. If a tree is associated with a reference data set, all tree versions belong to one set.

user rate type

Rate you enter at journal entry time to convert foreign currency transactions to your ledger currency.

value set

A set of valid values against which values entered by an end user are validated. The set may be tree structured (hierarchical).



