

**Oracle® Subledger Accounting**  
Implementation Guide  
Release 12.2  
**Part No. E48771-01**

September 2013

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 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, 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.

This software or hardware and documentation may provide access to or information on 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. 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.

---

# Contents

## Send Us Your Comments

## Preface

### 1 Subledger Accounting Options Setup

|   |     |
|---|-----|
| Subledger Accounting Options Setup Overview.....    | 1-1 |
| Subledger Accounting Options Setup Pages.....       | 1-1 |
| Setting Up Subledger Accounting Options.....        | 1-2 |
| Subledger Accounting Options Setup Description..... | 1-3 |

### 2 Accounting Methods Builder

|  |     |
|--|-----|
| Accounting Methods Builder (AMB) Overview.....         | 2-1 |
| Accounting Methods Builder Process.....                | 2-4 |
| Step 1. Define Journal Line Types.....                 | 2-5 |
| Step 2. Define Journal Entry Descriptions.....         | 2-5 |
| Step 3. Define Mapping Sets.....                       | 2-5 |
| Step 4. Define Account Derivation Rules.....           | 2-5 |
| Step 5. Define Supporting References.....              | 2-5 |
| Step 6. Define Journal Lines Definitions.....          | 2-6 |
| Step 7. Define Application Accounting Definitions..... | 2-6 |
| Step 8. Define Subledger Accounting Methods.....       | 2-6 |
| Copy and Modify Functionality.....                     | 2-6 |
| Transaction and Accounting Charts of Accounts.....     | 2-7 |
| Transaction Chart of Accounts.....                     | 2-7 |
| Accounting Chart of Accounts.....                      | 2-8 |
| Multiple Representations.....                          | 2-8 |

|   |             |
|---|-------------|
| Using the Transaction and Accounting Chart of Accounts.....         | 2-10        |
| Assigning the Transaction and Accounting Chart of Accounts.....     | 2-11        |
| Chart of Accounts Mapping.....                                      | 2-11        |
| <b>Event Model.....</b>   | <b>2-12</b> |
| <b>Custom Sources.....</b>  | <b>2-12</b> |
| Defining Custom Sources.....  | 2-13        |
| <b>Accounting Attributes Guideline.....</b>                         | <b>2-14</b> |
| <b>Post-Accounting Programs.....</b>                                | <b>2-15</b> |
| <b>Business Flows.....</b>  | <b>2-17</b> |
| Business Flows Features.....  | 2-19        |
| Business Flows Setup Considerations and Tasks.....                  | 2-21        |
| Copy from the Prior Journal Entry Business Flow Method Process..... | 2-22        |
| Copy from the Same Entry Business Flow Method Process.....          | 2-26        |
| <b>Journal Line Types.....</b>                                      | <b>2-29</b> |
| Journal Line Types Features.....                                    | 2-29        |
| Defining Journal Line Types.....                                    | 2-31        |
| To Define Journal Line Types.....                                   | 2-31        |
| Defining Conditions for Journal Line Types.....                     | 2-41        |
| Copying Journal Line Types and Their Associated Conditions.....     | 2-47        |
| <b>Defining Journal Entry Descriptions.....</b>                     | <b>2-47</b> |
| To Define a Journal Entry Description.....                          | 2-48        |
| To Define Journal Entry Description Details.....                    | 2-49        |
| Journal Entry Description Conditions.....                           | 2-51        |
| Copying Journal Entry Descriptions.....                             | 2-51        |
| <b>Mapping Sets.....</b>  | <b>2-51</b> |
| Defining Mapping Sets.....  | 2-52        |
| Mapping Sets Examples.....  | 2-55        |
| <b>Account Derivation Rules.....</b>                                | <b>2-57</b> |
| Accounting Flexfield Rules.....                                     | 2-58        |
| Segment Rules.....  | 2-59        |
| Value Set Rules.....  | 2-60        |
| Account Derivation Rules Examples.....                              | 2-61        |
| Defining Account Derivation Rules.....                              | 2-64        |
| Account Derivation Rules Conditions.....                            | 2-70        |
| <b>Supporting References.....</b>                                   | <b>2-72</b> |
| Defining Supporting References.....                                 | 2-76        |
| Updating Supporting References.....                                 | 2-77        |
| Duplicating Supporting References.....                              | 2-78        |
| Deleting Supporting References.....                                 | 2-78        |
| <b>Journal Lines Definitions.....</b>                               | <b>2-79</b> |
| Defining Journal Lines Definitions.....                             | 2-81        |

|   |       |
|---|-------|
| <b>Multiperiod Accounting.....</b>                            | 2-90  |
| Multiperiod Accounting Process.....                           | 2-90  |
| Multiperiod Accounting Example.....                           | 2-93  |
| Defining Multiperiod Accounting.....                          | 2-95  |
| <b>Accrual Reversals.....</b>                                 | 2-99  |
| <b>Application Accounting Definitions.....</b>                | 2-107 |
| Defining an Application Accounting Definition.....            | 2-108 |
| Copying and Modifying Application Accounting Definitions..... | 2-112 |
| Import Application Accounting Definitions.....                | 2-114 |
| <b>Export Application Accounting Definitions.....</b>         | 2-116 |
| <b>Subledger Accounting Methods.....</b>                      | 2-118 |
| To Define a Subledger Accounting Method.....                  | 2-119 |
| Copy and Modify a Subledger Accounting Method.....            | 2-120 |
| <b>Accounting Definitions Inquiry.....</b>                    | 2-121 |
| Accounting Definitions Inquiry Windows.....                   | 2-122 |

### **3 Create Accounting and Transfer Journal Entries to GL Programs**

|   |     |
|---|-----|
| <b>Overview of Create Accounting and Transfer Journal Entries to GL Programs.....</b> | 3-1 |
| Create Accounting Program.....  | 3-1 |
| Transfer Journal Entries to GL Program.....   | 3-4 |
| <b>Oracle Subledger Accounting Program Report.....</b>                                | 3-5 |
| <b>Transfer Journal Entries to GL Report.....</b>                                     | 3-5 |

### **4 Conversion of Historical Data for Reporting Currencies and Secondary Ledgers**

|   |     |
|---|-----|
| <b>Conversion of Historical Data for Reporting Currencies and Secondary Ledgers.....</b>      | 4-1 |
| <b>Running the SLA Secondary/ALC Ledger Historic Upgrade Program.....</b>                     | 4-2 |
| <b>Running the Upgrade Historical Subledger Transaction Accounting Program on Demand.....</b> | 4-4 |

### **5 Diagnostic Framework**

|  |     |
|--|-----|
| <b>Diagnostic Framework Overview.....</b>                      | 5-1 |
| <b>Diagnostic Framework Business Process Flow Diagram.....</b> | 5-1 |
| <b>Diagnostic Framework Features.....</b>                      | 5-3 |
| SLA: Enable Diagnostics Profile Option.....                    | 5-3 |
| Diagnostic Framework Execution.....                            | 5-3 |
| Transaction Objects Diagnostics Report.....                    | 5-3 |
| Purge Transaction Objects Diagnostics.....                     | 5-5 |

## **6 Subledger Journal Entries**

|  |     |
|--|-----|
| Subledger Journal Entries Overview.....        | 6-1 |
| Creating Manual Subledger Journal Entries..... | 6-2 |

## **7 Inquiries**

|  |     |
|--|-----|
| Subledger Accounting Inquiries Overview..... | 7-1 |
| Accounting Events Inquiry.....               | 7-1 |
| Subledger Journal Entry Headers Inquiry..... | 7-4 |
| Subledger Journal Entry Lines Inquiry.....   | 7-5 |
| Viewing Supporting Reference Balances.....   | 7-7 |

## **8 Reporting Sequence**

|  |     |
|--|-----|
| Reporting Sequence Overview.....             | 8-1 |
| Sequence Programs.....                       | 8-2 |
| Reporting Sequence Process Flow Diagram..... | 8-3 |
| Reporting Sequence Process Steps.....        | 8-4 |

## **9 Subledger Accounting Reports**

|   |      |
|---|------|
| Subledger Accounting Reports Overview.....            | 9-1  |
| Journal Entries Report.....                           | 9-3  |
| Account Analysis Report.....                          | 9-20 |
| Third Party Balances Report.....                      | 9-33 |
| Multiperiod Accounting Reports.....                   | 9-47 |
| Subledger Period Close Exceptions Report.....         | 9-62 |
| Subledger Accounting Rules Detail Listing Report..... | 9-66 |

## **10 Open Account Balances Listing**

|   |      |
|---|------|
| Open Account Balances Listing.....                      | 10-1 |
| Open Account Balances Listing Process Steps.....        | 10-1 |
| Creating Open Account Balances Listing Definitions..... | 10-2 |
| Open Account Balances Data Manager.....                 | 10-5 |
| Open Account Balances Listing.....                      | 10-5 |

## **11 Transaction Account Builder**

|   |      |
|---|------|
| Transaction Account Builder Overview.....       | 11-1 |
| Transaction Account Components.....             | 11-1 |
| Transaction Account Builder Setup Process ..... | 11-3 |

## **A Oracle Subledger Accounting Navigation Paths**

|                       |     |
|-----------------------|-----|
| Navigation Paths..... | A-1 |
|-----------------------|-----|

## **B Oracle Subledger Accounting Profile Options and Profile Option Categories**

|  |     |
|--|-----|
| Profile Options and Profile Option Categories Overview.....  | B-1 |
| Profile Option Category and Profile Option Descriptions..... | B-2 |
| Accounting Methods Builder Category.....                     | B-3 |
| Debug Category.....  | B-3 |
| Deployment Category.....                                     | B-6 |
| Security Category.....                                       | B-7 |

## **C Subledger Journal Entry Definition**

|  |      |
|--|------|
| Subledger Journal Entry Definition Overview.....   | C-1  |
| High Level Definition of a Subledger Journal Entry.....                                  | C-1  |
| Subledger Journal Entry Header Validation.....   | C-2  |
| Subledger Journal Entry Line Validation.....   | C-9  |
| Subledger Journal Entry Balancing Rules.....   | C-14 |
| Balanced by Ledger Currency.....   | C-15 |
| Balanced by Entered Currency.....  | C-15 |
| Balanced by Balancing Segment.....   | C-15 |
| Validation of Monetary Amounts (ENTERED_DR, ENTERED_CR, ACCOUNTED_DR, ACCOUNTED_CR)..... | C-16 |
| Example - Valid Debits and Credits.....  | C-19 |
| Example - Invalid Debits and Credits.....  | C-20 |

## **Index**



---

# Send Us Your Comments

**Oracle Subledger Accounting Implementation Guide, Release 12.2**

**Part No. E48771-01**

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

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

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

Note: Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the new Oracle E-Business Suite Release Online Documentation CD available on My Oracle Support and [www.oracle.com](http://www.oracle.com). It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: [appsdoc\\_us@oracle.com](mailto:appsdoc_us@oracle.com)

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

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

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



---

# Preface

## Intended Audience

Welcome to Release 12.2 of the *Oracle Subledger Accounting Implementation Guide*.

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

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

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

See Related Information Sources on page xii for more Oracle E-Business Suite product information.

## Documentation Accessibility

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>.

## Access to Oracle Support

Oracle customers 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.

## Structure

- 1 Subledger Accounting Options Setup
- 2 Accounting Methods Builder
- 3 Create Accounting and Transfer Journal Entries to GL Programs

- 4 Conversion of Historical Data for Reporting Currencies and Secondary Ledgers**
- 5 Diagnostic Framework**
- 6 Subledger Journal Entries**
- 7 Inquiries**
- 8 Reporting Sequence**
- 9 Subledger Accounting Reports**
- 10 Open Account Balances Listing**
- 11 Transaction Account Builder**
- A Oracle Subledger Accounting Navigation Paths**
- B Oracle Subledger Accounting Profile Options and Profile Option Categories**
- C Subledger Journal Entry Definition**

## Related Information Sources

This book is included in the Oracle E-Business Suite Documentation Library, which is supplied in the Release 12.2 Media Pack. If this guide refers you to other Oracle E-Business Suite documentation, use only the latest Release 12.2 versions of those guides.

### Online Documentation

All Oracle E-Business Suite documentation is available online (HTML or PDF).

- **Online Help** - Online help patches (HTML) are available on My Oracle Support.
- **PDF Documentation** - See the Oracle E-Business Suite Documentation Library for current PDF documentation for your product with each release.
- **Release Notes** - For information about changes in this release, including new features, known issues, and other details, see the release notes for the relevant product, available on My Oracle Support.
- **Oracle Electronic Technical Reference Manual** - The Oracle Electronic Technical Reference Manual (eTRM) contains database diagrams and a detailed description of database tables, forms, reports, and programs for each Oracle E-Business Suite product. This information helps you convert data from your existing applications and integrate Oracle E-Business Suite data with non-Oracle applications, and write custom reports for Oracle E-Business Suite products. The Oracle eTRM is available on My Oracle Support.

### Related Guides

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

#### Oracle Alert User's Guide

This guide explains how to define periodic and event alerts to monitor the status of your Oracle E-Business Suite data.

#### Oracle Application Framework Developer's Guide

This guide contains the coding standards followed by the Oracle E-Business Suite development staff to produce applications built with Oracle Application Framework. This guide is available in PDF format on My Oracle Support and as online documentation in JDeveloper 10g with Oracle Application Extension.

### **Oracle Application Framework Personalization Guide**

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

### **Oracle Fusion Middleware Adapter for Oracle Applications User's Guide (Oracle Application Server Adapter for Oracle Applications User's Guide)**

This guide covers the use of Adapter for Oracle Applications in developing integrations between Oracle E-Business Suite and trading partners.

Please note that the user's guide can be found in the following documentation libraries:

- As part of the Oracle Fusion Middleware and SOA Suite in 11g, *Oracle Fusion Middleware Adapter for Oracle Applications User's Guide* is available in the Oracle Fusion Middleware 11g Documentation Library.
- As part of the Oracle Application Server in 10g, *Oracle Application Server Adapter for Oracle Applications User's Guide* is available in the Oracle Application Server 10g Documentation Library.

### **Oracle Diagnostics Framework User's Guide**

This manual contains information on implementing and administering diagnostics tests for Oracle E-Business Suite using the Oracle Diagnostics Framework.

### **Oracle E-Business Suite Concepts**

This book is intended for all those planning to deploy Oracle E-Business Suite Release 12.2, or contemplating significant changes to a configuration. After describing the Oracle E-Business Suite architecture and technology stack, it focuses on strategic topics, giving a broad outline of the actions needed to achieve a particular goal, plus the installation and configuration choices that may be available.

### **Oracle E-Business Suite CRM System Administrator's Guide**

This manual describes how to implement the CRM Technology Foundation (JTT) and use its System Administrator Console.

### **Oracle E-Business Suite Desktop Integration Framework Developer's Guide**

Oracle E-Business Suite Desktop Integration Framework is a development tool that lets you define custom integrators for use with Oracle Web Applications Desktop Integrator. This guide describes how to define and manage integrators and all associated supporting objects, as well as how to download and upload integrator definitions.

### **Oracle E-Business Suite Developer's Guide**

This guide contains the coding standards followed by the Oracle E-Business Suite

development staff. It describes the Oracle Application Object Library components needed to implement the Oracle E-Business Suite user interface described in the *Oracle E-Business Suite User Interface Standards for Forms-Based Products*. It provides information to help you build your custom Oracle Forms Developer forms so that they integrate with Oracle E-Business Suite. In addition, this guide has information for customizations in features such as concurrent programs, flexfields, messages, and logging.

### **Oracle E-Business Suite Flexfields Guide**

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

### **Oracle E-Business Suite Installation Guide: Using Rapid Install**

This book is intended for use by anyone who is responsible for installing or upgrading Oracle E-Business Suite. It provides instructions for running Rapid Install either to carry out a fresh installation of Oracle E-Business Suite Release 12.2, or as part of an upgrade to Release 12.2.

### **Oracle E-Business Suite Maintenance Guide**

This guide contains information about the strategies, tasks, and troubleshooting activities that can be used to help ensure an Oracle E-Business Suite system keeps running smoothly, together with a comprehensive description of the relevant tools and utilities. It also describes how to patch a system, with recommendations for optimizing typical patching operations and reducing downtime.

### **Oracle E-Business Suite Security Guide**

This guide contains information on a comprehensive range of security-related topics, including access control, user management, function security, data security, and auditing. It also describes how Oracle E-Business Suite can be integrated into a single sign-on environment.

### **Oracle E-Business Suite Setup Guide**

This guide contains information on system configuration tasks that are carried out either after installation or whenever there is a significant change to the system. The activities described include defining concurrent programs and managers, enabling Oracle Applications Manager features, and setting up printers and online help.

### **Oracle E-Business Suite User's Guide**

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

### **Oracle E-Business Suite User Interface Standards for Forms-Based Products**

This guide contains the user interface (UI) standards followed by the Oracle E-Business Suite development staff. It describes the UI for the Oracle E-Business Suite products and how to apply this UI to the design of an application built by using Oracle Forms.

### **Oracle E-Business Suite Integrated SOA Gateway User's Guide**

This guide describes the high level service enablement process, explaining how users can browse and view the integration interface definitions and services residing in Oracle Integration Repository.

### **Oracle E-Business Suite Integrated SOA Gateway Implementation Guide**

This guide explains how integration repository administrators can manage and administer the Web service activities for integration interfaces including native packaged integration interfaces, composite services (BPEL type), and custom integration interfaces. It also describes how to invoke Web services from Oracle E-Business Suite by employing the Oracle Workflow Business Event System, and how to manage Web service security, configure logs, and monitor SOAP messages.

### **Oracle E-Business Suite Integrated SOA Gateway Developer's Guide**

This guide describes how system integration developers can perform end-to-end service integration activities. These include orchestrating discrete Web services into meaningful end-to-end business processes using business process execution language (BPEL), and deploying BPEL processes at run time.

This guide also explains how to invoke Web services using the Service Invocation Framework. This includes defining Web service invocation metadata, invoking Web services, and testing the Web service invocation.

### **Oracle e-Commerce Gateway User's Guide**

This guide describes the functionality of Oracle e-Commerce Gateway and the necessary setup steps in order for Oracle E-Business Suite to conduct business with trading partners through Electronic Data Interchange (EDI). It also describes how to run extract programs for outbound transactions, import programs for inbound transactions, and the relevant reports.

### **Oracle e-Commerce Gateway Implementation Guide**

This guide describes implementation details, highlighting additional setup steps needed for trading partners, code conversion, and Oracle E-Business Suite. It also provides architecture guidelines for transaction interface files, troubleshooting information, and a description of how to customize EDI transactions.

### **Oracle iSetup Developer's Guide**

This manual describes how to build, test, and deploy Oracle iSetup Framework interfaces.

### **Oracle iSetup User's Guide**

This guide describes how to use Oracle iSetup to migrate data between different instances of the Oracle E-Business Suite and generate reports. It also includes configuration information, instance mapping, and seeded templates used for data migration.

### **Oracle Report Manager User's Guide**

Oracle Report Manager is an online report distribution system that provides a secure and centralized location to produce and manage point-in-time reports. Oracle Report Manager users can be either report producers or report consumers. Use this guide for information on setting up and using Oracle Report Manager.

### **Oracle Web Applications Desktop Integrator Implementation and Administration Guide**

Oracle Web Applications Desktop Integrator brings Oracle E-Business Suite functionality to a spreadsheet, where familiar data entry and modeling techniques can be used to complete Oracle E-Business Suite tasks. You can create formatted spreadsheets on your desktop that allow you to download, view, edit, and create Oracle E-Business Suite data, which you can then upload. This guide describes how to implement Oracle Web Applications Desktop Integrator and how to define mappings, layouts, style sheets, and other setup options.

### **Oracle Workflow Administrator's Guide**

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

### **Oracle Workflow Developer's Guide**

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

### **Oracle Workflow User's Guide**

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

### **Oracle Workflow API Reference**

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

### **Oracle Workflow Client Installation Guide**

This guide describes how to install the Oracle Workflow Builder and Oracle XML Gateway Message Designer client components for Oracle E-Business Suite.

### **Oracle XML Gateway User's Guide**

This guide describes Oracle XML Gateway functionality and each component of the Oracle XML Gateway architecture, including Message Designer, Oracle XML Gateway Setup, Execution Engine, Message Queues, and Oracle Transport Agent. It also explains how to use Collaboration History that records all business transactions and messages exchanged with trading partners.

The integrations with Oracle Workflow Business Event System, and the Business-to-Business transactions are also addressed in this guide.

## **Oracle XML Publisher Report Designer's Guide**

Oracle XML Publisher is a template-based reporting solution that merges XML data with templates in RTF or PDF format to produce a variety of outputs to meet a variety of business needs. Using Microsoft Word or Adobe Acrobat as the design tool, you can create pixel-perfect reports from the Oracle E-Business Suite. Use this guide to design your report layouts.

This guide is available through the Oracle E-Business Suite online help.

## **Oracle XML Publisher Administration and Developer's Guide**

Oracle XML Publisher is a template-based reporting solution that merges XML data with templates in RTF or PDF format to produce a variety of outputs to meet a variety of business needs. Outputs include: PDF, HTML, Excel, RTF, and eText (for EDI and EFT transactions). Oracle XML Publisher can be used to generate reports based on existing Oracle E-Business Suite report data, or you can use Oracle XML Publisher's data extraction engine to build your own queries. Oracle XML Publisher also provides a robust set of APIs to manage delivery of your reports via e-mail, fax, secure FTP, printer, WebDav, and more. This guide describes how to set up and administer Oracle XML Publisher as well as how to use the Application Programming Interface to build custom solutions.

This guide is available through the Oracle E-Business Suite online help.

## **Oracle E-Business Suite Upgrade Guide: Release 12.0 and 12.1 to 12.2:**

This guide provides information for DBAs and Applications Specialists who are responsible for upgrading Release 12.0 and 12.1 Oracle E-Business Suite system (techstack and products) to Release 12.2. In addition to information about applying the upgrade driver, it outlines pre-upgrade steps and post-upgrade steps, and provides descriptions of product-specific functional changes and suggestions for verifying the upgrade and reducing downtime.

## **Oracle Advanced Global Intercompany System User's Guide:**

This guide describes the self service application pages available for Intercompany users. It includes information on setting up intercompany, entering intercompany transactions, importing transactions from external sources and generating reports.

## **Oracle Advanced Collections User Guide:**

This guide describes how to use the features of Oracle Advanced Collections to manage your collections activities. It describes how collections agents and managers can use Oracle Advanced Collections to identify delinquent customers, review payment history and aging data, process payments, use strategies and dunning plans to automate the collections process, manage work assignments, and handle later-stage delinquencies.

## **Oracle Advanced Collections Implementation Guide:**

This guide describes how to configure Oracle Advanced Collections and its integrated products. It contains the steps required to set up and verify your implementation of Oracle Advanced Collections.

### **Oracle Assets User Guide:**

This guide provides you with information on how to implement and use Oracle Assets. Use this guide to understand the implementation steps required for application use, including defining depreciation books, depreciation method, and asset categories. It also contains information on setting up assets in the system, maintaining assets, retiring and reinstating assets, depreciation, group depreciation, accounting and tax accounting, budgeting, online inquiries, impairment processing, and Oracle Assets reporting. The guide explains using Oracle Assets with Multiple Reporting Currencies (MRC). This guide also includes a comprehensive list of profile options that you can set to customize application behavior.

### **Oracle Bill Presentment Architecture User's Guide:**

This guide provides you information on using Oracle Bill Presentment Architecture. Consult this guide to create and customize billing templates, assign a template to a rule and submit print requests. This guide also provides detailed information on page references, seeded content items and template assignment attributes.

### **Oracle Cash Management User Guide:**

This guide describes how to use Oracle Cash Management to clear your receipts, as well as reconcile bank statements with your outstanding balances and transactions. This manual also explains how to effectively manage and control your cash cycle. It provides comprehensive bank reconciliation and flexible cash forecasting.

### **Oracle Credit Management User Guide:**

This guide provides you with information on how to use Oracle Credit Management. This guide includes implementation steps, such as how to set up credit policies, as well as details on how to use the credit review process to derive credit recommendations that comply with your credit policies. This guide also includes detailed information about the public application programming interfaces (APIs) that you can use to extend Oracle Credit Management functionality.

### **Oracle Customer Data Librarian User Guide:**

This guide describes how to use Oracle Customer Data Librarian to establish and maintain the quality of the Trading Community Architecture Registry, focusing on consolidation, cleanliness, and completeness. Oracle Customer Data Librarian has all of the features in Oracle Customers Online, and is also part of the Oracle Customer Data Management product family.

### **Oracle Customer Data Librarian Implementation Guide:**

This guide describes how to implement Oracle Customer Data Librarian. As part of implementing Oracle Customer Data Librarian, you must also complete all the implementation steps for Oracle Customers Online.

### **Oracle Customers Online User Guide:**

This guide describes how to use Oracle Customers Online to view, create, and maintain your customer information. Oracle Customers Online is based on Oracle Trading

Community Architecture data model and functionality, and is also part of the Oracle Customer Data Management product family.

**Oracle Customers Online Implementation Guide:**

This guide describes how to implement Oracle Customers Online.

**Oracle E-Business Suite Multiple Organizations Implementation Guide:**

This guide describes the multiple organizations concepts in Oracle E-Business Suite. It describes in detail on setting up and working effectively with multiple organizations in Oracle E-Business Suite.

**Oracle E-Business Tax User Guide:**

This guide describes the entire process of setting up and maintaining tax configuration data, as well as applying tax data to the transaction line. It describes the entire regime-to-rate setup flow of tax regimes, taxes, statuses, rates, recovery rates, tax jurisdictions, and tax rules. It also describes setting up and maintaining tax reporting codes, fiscal classifications, tax profiles, tax registrations, configuration options, and third party service provider subscriptions. You also use this manual to maintain migrated tax data for use with E-Business Tax.

**Oracle E-Business Tax Implementation Guide:**

This guide provides a conceptual overview of the E-Business Tax tax engine, and describes the prerequisite implementation steps to complete in other applications in order to set up and use E-Business Tax. The guide also includes extensive examples of setting up country-specific tax requirements.

**Oracle E-Business Tax Reporting Guide:**

This guide explains how to run all tax reports that make use of the E-Business Tax data extract. This includes the Tax Reporting Ledger and other core tax reports, country-specific VAT reports, and Latin Tax Engine reports.

**Oracle E-Business Tax: Vertex Q-Series and Taxware Sales/Use Tax System Implementation Guide**

This guide explains how to setup and use the services of third party tax service providers for US Sales and Use tax. The tax service providers are Vertex Q-Series and Taxware Sales/Use Tax System. When implemented, the Oracle E-Business Tax service subscription calls one of these tax service providers to return a tax rate or amount whenever US Sales and Use tax is calculated by the Oracle E-Business Tax tax engine. This guide provides setup steps, information about day-to-day business processes, and a technical reference section.

**Oracle Embedded Data Warehouse User Guide:**

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

**Oracle Embedded Data Warehouse Implementation Guide:**

This guide describes how to implement Embedded Data Warehouse, including how to

set up the intelligence areas.

**Oracle Embedded Data Warehouse Install Guide:**

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

**Oracle Financial Accounting Hub Implementation Guide:**

This guide provides detailed implementation information that leverages the features of Oracle Subledger Accounting to generate accounting.

**Oracle Financial Services Reference Guide:**

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

**Oracle Financial Services Implementation Guide:**

This guide describes how to set up Oracle Financial Services applications in Release 12.

**Oracle Financial Services Reporting Administration Guide:**

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

**Oracle Financials and Oracle Procurement Functional Upgrade Guide: Release 11i to Release 12:**

This guide provides detailed information about the functional impacts of upgrading Oracle Financials and Oracle Procurement products from Release 11i to Release 12. This guide supplements the *Oracle E-Business Suite Upgrade Guide: Release 12.0 and 12.1 to 12.2*.

**Oracle Financials Concepts Guide:**

This guide describes the fundamental concepts of Oracle Financials. The guide is intended to introduce readers to the concepts used in the applications, and help them compare their real world business, organization, and processes to those used in the applications.

**Oracle Financials Country-Specific Installation Supplement:**

This guide provides general country information, such as responsibilities and report security groups, as well as any post-install steps required by some countries.

**Oracle Financials for the Americas User Guide:**

This guide describes functionality developed to meet specific business practices in countries belonging to the Americas region. Consult this user guide along with your financial product user guides to effectively use Oracle Financials in your country.

**Oracle Financials for Asia/Pacific User Guide:**

This guide describes functionality developed to meet specific business practices in countries belonging to the Asia/Pacific region. Consult this user guide along with your

financial product user guides to effectively use Oracle Financials in your country.

**Oracle Financials for Europe User Guide:**

This guide describes functionality developed to meet specific business practices in countries belonging to the European region. Consult this user guide along with your financial product user guides to effectively use Oracle Financials in your country.

**Oracle Financials for India User's Guide:**

This guide provides information on how to use Oracle Financials for India. Use this guide to learn how to create and maintain setup related to India taxes, defaulting and calculation of taxes on transactions. This guide also includes information about accounting and reporting of taxes related to India.

**Oracle Financials for India Implementation Guide:**

This guide provides information on how to implement Oracle Financials for India. Use this guide to understand the implementation steps required for application use, including how to set up taxes, tax defaulting hierarchies, set up different tax regimes, organization and transactions.

**Oracle Financials Glossary:**

The glossary includes definitions of common terms that are shared by all Oracle Financials products. In some cases, there may be different definitions of the same term for different Financials products. If you are unsure of the meaning of a term you see in an Oracle Financials guide, please refer to the glossary for clarification. You can find the glossary in the online help or in the *Oracle Financials Implementation Guide*.

**Oracle Financials Implementation Guide:**

This guide provides information on how to implement the Oracle Financials E-Business Suite. It guides you through setting up your organizations, including legal entities, and their accounting, using the Accounting Setup Manager. It covers intercompany accounting and sequencing of accounting entries, and it provides examples.

**Oracle Financials RXi Reports Administration Tool User Guide:**

This guide describes how to use the RXi reports administration tool to design the content and layout of RXi reports. RXi reports let you order, edit, and present report information to better meet your company's reporting needs.

**Oracle General Ledger Implementation Guide:**

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

**Oracle General Ledger Reference Guide**

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

**Oracle General Ledger User's Guide:**

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

**Oracle Incentive Compensation Implementation Guide:**

This guide provides Compensation Administrators with guidance during implementation of Oracle Incentive Compensation. The procedures are presented in the recommended order that they should be performed for successful implementation. Appendixes are included that describe system profiles, lookups, and other useful information.

**Oracle Incentive Compensation User Guide:**

This guide helps Compensation Managers, Compensation Analysts, and Plan administrators to manage Oracle Incentive Compensation on a day-to-day basis. Learn how to create and manage rules hierarchies, create compensation plans, collect transactions, calculate and pay commission, and use Sales Credit Allocation.

**Oracle Internet Expenses Implementation and Administration Guide:**

This book explains in detail how to configure Oracle Internet Expenses and describes its integration with other applications in the E-Business Suite, such as Oracle Payables and Oracle Projects. Use this guide to understand the implementation steps required for application use, including how to set up policy and rate schedules, credit card policies, audit automation, and the expenses spreadsheet. This guide also includes detailed information about the client extensions that you can use to extend Oracle Internet Expenses functionality.

**Oracle iAssets User Guide**

This guide provides information on how to implement and use Oracle iAssets. Use this guide to understand the implementation steps required for application use, including setting up Oracle iAssets rules and related product setup steps. It explains how to define approval rules to facilitate the approval process. It also includes information on using the Oracle iAssets user interface to search for assets, create self-service transfer requests and view notifications.

**Oracle iProcurement Implementation and Administration Guide:**

This manual describes how to set up and administer Oracle iProcurement. Oracle iProcurement enables employees to requisition items through a self-service, Web interface.

**Oracle iReceivables Implementation Guide:**

This guide provides information on how to implement Oracle iReceivables. Use this guide to understand the implementation steps required for application use, including how to set up and configure iReceivables, and how to set up the Credit Memo Request workflow. There is also a chapter that provides an overview of major features available in iReceivables.

**Oracle iSupplier Portal User Guide:**

This guide contains information on how to use Oracle iSupplier Portal to enable secure transactions between buyers and suppliers using the Internet. Using Oracle iSupplier Portal, suppliers can monitor and respond to events in the procure-to-pay cycle.

**Oracle iSupplier Portal Implementation Guide:**

This guide contains information on how to implement Oracle iSupplier Portal and enable secure transactions between buyers and suppliers using the Internet.

**Oracle Loans User Guide:**

This guide describes how to set up and use Oracle Loans. It includes information on how to create, approve, fund, amortize, bill, and service extended repayment plan and direct loans.

**Oracle Partner Management Implementation and Administration Guide:**

This guide helps Vendor administrators to set up and maintain relationships and programs in the Partner Management application. The main areas include setting up the partner and channel manager dashboards, partner setup, partner programs and enrollment, opportunity and referral management, deal registration, special pricing management, and partner fund management.

**Oracle Partner Management Vendor User Guide:**

This guide assists vendor users in using Partner Management on a daily basis. This includes interaction with the partner and channel manager dashboards, working with partners and partner programs, managing opportunities and referrals, registering deals, and working with special pricing and partner funds.

**Oracle Payables User's Guide:**

This guide describes how to use Oracle Payables to create invoices and make payments. In addition, it describes how to enter and manage suppliers, import invoices using the Payables open interface, manage purchase order and receipt matching, apply holds to invoices, and validate invoices. It contains information on managing expense reporting, procurement cards, and credit cards. This guide also explains the accounting for Payables transactions.

**Oracle Payables Implementation Guide:**

This guide provides you with information on how to implement Oracle Payables. Use this guide to understand the implementation steps required for how to set up suppliers, payments, accounting, and tax.

**Oracle Payables Reference Guide:**

This guide provides you with detailed information about the Oracle Payables open interfaces, such as the Invoice open interface, which lets you import invoices. It also includes reference information on purchase order matching and purging purchasing information.

**Oracle Payments Implementation Guide:**

This guide describes how Oracle Payments, as the central payment engine for the

Oracle E-Business Suite, processes transactions, such as invoice payments from Oracle Payables, bank account transfers from Oracle Cash Management, and settlements against credit cards and bank accounts from Oracle Receivables. This guide also describes how Oracle Payments is integrated with financial institutions and payment systems for receipt and payment processing, known as funds capture and funds disbursement, respectively. Additionally, the guide explains to the implementer how to plan the implementation of Oracle Payments, how to configure it, set it up, test transactions, and how use it with external payment systems.

**Oracle Payments User's Guide:**

This guide describes how Oracle Payments, as the central payment engine for the Oracle E-Business Suite, processes transactions, such as invoice payments from Oracle Payables, bank account transfers from Oracle Cash Management, and settlements against credit cards and bank accounts from Oracle Receivables. This guide also describes to the Payment Administrator how to monitor the funds capture and funds disbursement processes, as well as how to remedy any errors that may arise.

**Oracle Procurement Buyer's Guide to Punchout and Transparent Punchout:**

This guide contains necessary information for customers implementing remote catalog content on a supplier's Web site or on Oracle Exchange.

**Oracle Procurement Contracts Online Help:**

This guide is provided as online help only from the Oracle Procurement Contracts application and includes information about creating and managing your contract terms library.

**Oracle Procurement Contracts Implementation and Administration Guide:**

This guide describes how to set up and administer Oracle Procurement Contracts. Oracle Procurement Contracts enables employees to author and maintain complex contracts through a self-service, Web interface.

**Oracle Public Sector Financials User Guide:**

This guide describes how to set up and administer Oracle Public Sector Advanced Features. It describes Encumbrance Reconciliation Reports, GASB 34/35 Asset Accounting, and Funds Available Enhancements.

**Oracle Purchasing User's Guide:**

This guide describes how to create and approve purchasing documents, including requisitions, different types of purchase orders, quotations, RFQs, and receipts. This guide also describes how to manage your supply base through agreements, sourcing rules, and approved supplier lists. In addition, this guide explains how you can automatically create purchasing documents based on business rules through integration with Oracle Workflow technology, which automates many of the key procurement processes.

**Oracle Receivables User Guide:**

This guide provides you with information on how to use Oracle Receivables. Use this

guide to learn how to create and maintain transactions and bills receivable, enter and apply receipts, enter customer information, and manage revenue. This guide also includes information about accounting in Receivables. Use the Standard Navigation Paths appendix to find out how to access each Receivables window.

**Oracle Receivables Implementation Guide:**

This guide provides you with information on how to implement Oracle Receivables. Use this guide to understand the implementation steps required for application use, including how to set up customers, transactions, receipts, accounting, tax, and collections. This guide also includes a comprehensive list of profile options that you can set to customize application behavior.

**Oracle Receivables Reference Guide:**

This guide provides you with detailed information about all public application programming interfaces (APIs) that you can use to extend Oracle Receivables functionality. This guide also describes the Oracle Receivables open interfaces, such as AutoLockbox which lets you create and apply receipts and AutoInvoice which you can use to import and validate transactions from other systems. Archiving and purging Receivables data is also discussed in this guide.

**Oracle Sourcing Implementation and Administration Guide:**

This guide contains information on how to implement Oracle Sourcing to enable participants from multiple organizations to exchange information, conduct bid and auction processes, and create and implement buying agreements. This allows professional buyers, business experts, and suppliers to participate in a more agile and accurate sourcing process.

**Oracle Subledger Accounting Implementation Guide:**

This guide provides setup information for Oracle Subledger Accounting features, including the Accounting Methods Builder. You can use the Accounting Methods Builder to create and modify the setup for subledger journal lines and application accounting definitions for Oracle subledger applications. This guide also discusses the reports available in Oracle Subledger Accounting and describes how to inquire on subledger journal entries.

**Oracle Supplier Scheduling User's Guide:**

This guide describes how you can use Oracle Supplier Scheduling to calculate and maintain planning and shipping schedules and communicate them to your suppliers.

**Oracle iProcurement Implementation and Administration Guide:**

This manual describes how to set up and administer Oracle iProcurement. Oracle iProcurement enables employees to requisition items through a self-service, Web interface.

**Oracle Procurement Contracts Implementation and Administration Guide:**

This manual describes how to set up and administer Oracle Procurement Contracts. Oracle Procurement Contracts enables employees to author and maintain complex

contracts through a self-service, Web interface.

**Oracle Trading Community Architecture User Guide:**

This guide describes the Oracle Trading Community Architecture (TCA) and how to use features from the Trading Community Manager responsibility to create, update, enrich, and cleanse the data in the TCA Registry. It also describes how to use Resource Manager to define and manage resources.

**Oracle Trading Community Architecture Administration Guide:**

This guide describes how to administer and implement Oracle Trading Community Architecture (TCA). You set up, control, and manage functionality that affects data in the TCA Registry. It also describes how to set up and use Resource Manager to manage resources.

**Oracle Trading Community Architecture Reference Guide:**

This guide contains seeded relationship types, seeded Data Quality Management data, D&B data elements, Bulk Import interface table fields and validations, and a comprehensive glossary. This guide supplements the documentation for Oracle Trading Community Architecture and all products in the Oracle Customer Data Management family.

**Oracle Trading Community Architecture Technical Implementation Guide:**

This guide explains how to use the public Oracle Trading Community Architecture application programming interfaces (APIs) and develop callouts based on Oracle Workflow Business Events System (BES). For each API, this guide provides a description of the API, the PL/SQL procedure, and the Java method, as well as a table of the parameter descriptions and validations. For each BES callout, this guide provides the name of the logical entity, its description, and the ID parameter name. Also included are setup instructions and sample code.

**Oracle U.S. Federal Financials User's Guide:**

This guide describes the common concepts for an integrated financial management solution for federal agencies to comply with the requirements of the U.S. Federal government. It describes the product architecture and provides information on Budget Execution, Prompt Payment, Treasury payments, Third party payments, Interagency transactions, Receivables management, Federal reports, CCR Integration, and Year End Closing.

**Oracle U.S. Federal Financials Implementation Guide:**

This guide describes the common concepts for an integrated financial management solution for federal agencies. It includes a consolidated setup checklist by page and provides detailed information on how to set up, maintain, and troubleshoot the Federal Financial application for the following functional areas: Sub Ledger Accounting, Budget Execution, Prompt Payment, Treasury payments, Third party payments, Interagency transactions, Receivables management, Federal reports, CCR Integration, and Year End Closing.

## **Oracle Projects Documentation Set**

### **Oracle Projects Implementation Guide:**

Use this guide to implement Oracle Projects. This guide also includes appendixes covering function security, menus and responsibilities, and profile options.

### **Oracle Project Costing User Guide:**

Use this guide to learn detailed information about Oracle Project Costing. Oracle Project Costing provides the tools for processing project expenditures, including calculating their cost to each project and determining the General Ledger accounts to which the costs are posted.

### **Oracle Project Billing User Guide:**

This guide shows you how to use Oracle Project Billing to define revenue and invoicing rules for your projects, generate revenue, create invoices, and integrate with other Oracle Applications to process revenue and invoices, process client invoicing, and measure the profitability of your contract projects.

### **Oracle Project Management User Guide:**

This guide shows you how to use Oracle Project Management to manage projects through their lifecycles - from planning, through execution, to completion.

### **Oracle Project Portfolio Analysis User Guide:**

This guide contains the information you need to understand and use Oracle Project Portfolio Analysis. It includes information about project portfolios, planning cycles, and metrics for ranking and selecting projects for a project portfolio.

### **Oracle Project Resource Management User Guide:**

This guide provides you with information on how to use Oracle Project Resource Management. It includes information about staffing, scheduling, and reporting on project resources.

## **Oracle Grants Accounting Documentation**

### **Oracle Grants Accounting User Guide:**

This guide provides you with information about how to implement and use Oracle Grants Accounting. Use this guide to understand the implementation steps required for application use, including defining award types, award templates, allowed cost schedules, and burden set up. This guide also explains how to use Oracle Grants Accounting to track grants and funded projects from inception to final reporting.

## **Oracle Property Manager Documentation**

### **Oracle Property Manager User Guide:**

Use this guide to learn how to use Oracle Property Manager to create and administer properties, space assignments, and lease agreements.

### **Oracle Property Manager Implementation Guide:**

Use this guide to learn how to implement Oracle Property Manager and perform basic setup steps such as setting system options and creating lookup codes, contacts, milestones, grouping rules, term templates, and a location hierarchy. This guide also describes the setup steps that you must complete in other Oracle applications before you can use Oracle Property Manager.

## Integration Repository

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

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

You can navigate to the Oracle Integration Repository through Oracle E-Business Suite Integrated SOA Gateway.

## Do Not Use Database Tools to Modify Oracle E-Business Suite Data

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

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

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

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

---

# Subledger Accounting Options Setup

## Subledger Accounting Options Setup Overview

Subledger accounting options define how journal entries are generated from subledger transactions at the subledger application level. These options are set up for the primary and secondary ledgers only. Set up ledgers in General Ledger's Accounting Setup Manager.

See: Creating Accounting Setups, *Oracle Financials Implementation Guide*

All subledgers assigned to a ledger inherit the subledger accounting method from that ledger.

## Related Topics

For detailed implementation information on leveraging the features of Oracle Subledger Accounting to generate accounting for front-office banking and other financial services source systems, see: Overview of the Financial Services Accounting Hub for Implementers, *Oracle Financial Services Accounting Hub Implementation Guide*

## Subledger Accounting Options Setup Pages

The Subledger Accounting options setup pages are as follows:

### 1. Subledger Applications

Access this page from the Accounting Setups page in General Ledger. This page is displayed in the context of a primary or secondary ledger. All subledger applications registered with Subledger Accounting are displayed. In this page, select the application to update and the Update Subledger Accounting Options page opens. You can choose to disable Subledger Accounting for a particular application in the Update Accounting Options page of a secondary ledger.

See: Accounting Setups Page, *Oracle Financials Implementation Guide*

**Note:** If you have a ledger setup and want to add an event class, run the Subledger Accounting options setup concurrent program.

## 2. Update Accounting Options

This page displays the Subledger Accounting options for the ledger and the subledger application. Review and update the options.

The view for this page depends on the subledger application type (valuation method or nonvaluation method) and the ledger type (primary or secondary). The views are as follows:

- Launch defaults and ledger options view that displays the accounting program defaults and the event class options for the following:
  - A primary ledger for any subledger application that is a valuation method type or a nonvaluation method type
  - A secondary ledger for a valuation method subledger application
- Ledger options view that displays the event class options for the secondary ledger for a non-valuation method subledger application

## 3. Update System Options

You can define the processing unit size at the event class level to process a large number of items in one commit cycle. A processing unit is the number of transactions processed by the Create Accounting program in one commit cycle. The Create Accounting program processes the default processing unit size at the application level.

# Setting Up Subledger Accounting Options

## Prerequisites

- Set up a subledger application in the Accounting Methods Builder (AMB).  
See: Accounting Methods Builder (AMB) Introduction, page 2-1
- Set up ledgers in General Ledger.  
See: Primary Ledger Setup Steps, *Oracle Financials Implementation Guide* and Secondary Ledger Setup Steps, *Oracle Financials Implementation Guide*

## To Update Subledger Accounting Options and System Options

The screenshot shows the Oracle Accounting Setup Flow interface. The top navigation bar includes Home, Logout, Preferences, and Diagnostics, with Configuration selected. The breadcrumb navigation shows Configuration > Configuration: Big Apple, Corp Configuration > Subledger Applications > Update Accounting Options: Receivables > Update Accounting Options: SLA San Francisco Bay Area. The main content area is titled 'Update Accounting Options: SLA San Francisco Bay Area'. It contains two sections: 'Accounting Program Defaults' and 'Event Class Options'. The 'Accounting Program Defaults' section includes fields for General Ledger Journal Entry Summarization (Summarize by GL Period), Reversal Method (Switch DR/CR), Accounting Program Mode (Final), Accounting Report Level (Summary), Allow Mode Override (Yes), Allow Report Override (Yes), Transfer to GL (Yes), Stop at Error Limit (No), Allow Transfer Override (Yes), Post in GL (No), and Allow Post Override (Yes). The 'Event Class Options' section allows overriding default journal categories for event classes: Bonds (Manual Adjustments), SCIRS and CCIRS (Manual Adjustments), and Manual (Other). Both sections have 'Cancel' and 'Apply' buttons at the bottom.

1. In the Accounting Setup Manager, execute a search for a ledger.
2. In the Results region, click the Update Accounting Options icon for a ledger.
3. In the Primary Ledger region or the Secondary Ledger region, click the Update icon for Subledger Accounting Options.
4. In the Subledger Applications page, click the Update Accounting Options icon to update accounting options or the Update System Options icon to define the processing unit size.
5. Update the accounting options as described in the following section, Subledger Accounting Options Setup Description, page 1-3.

## Subledger Accounting Options Setup Description

### General Options Region

#### Subledger Accounting ENABLED

The Subledger Accounting option is visible only for secondary ledgers in the ledger options view described in Subledger Accounting Options Setup Pages, page 1-1. Enable or disable the subledger application for the ledger. No entries will be generated by Subledger Accounting for an application if Subledger Accounting is disabled for the ledger.

#### General Ledger Journal Entry Summarization

The General Ledger Journal Entry Summarization option determines whether subledger journal entries are summarized when they are transferred to General Ledger as described in the table below.

***General Ledger Journal Entry Summarization Options***

| Option                 | Description   |
|------------------------|---|
| Summarize by GL Period | <p>Default option; indicates that all subledger entry lines with the same GL period, General Ledger journal category, account, entered currency, side, and balance type are summarized into a single General Ledger entry; disabled if ledger uses daily balancing</p> <p>The GL effective date defaults to the last date of the accounting period.</p> |
| Summarize by GL Date   | <p>Indicates that all subledger entry lines with the same GL date, General Ledger journal category, account, entered currency, side, and balance type are summarized into a general ledger entry; default value if Summarize by GL period is disabled</p> <p>The GL effective date is equal to the subledger GL date.</p>                               |
| No Summarization       | <p>Indicates that the subledger journal entries in Subledger Accounting and General Ledger will have the same level of detail</p> <p>The GL effective date is equal to the subledger GL date.</p>   |

The table below describes the impact of the summarization options based on whether the journal line type is Summary or Detail.

***Impact of Summarization Option***

| Option                 | Journal Line Type - Summary                                  | Journal Line Type - Detail                                       |
|------------------------|--|--|
| Summarize by GL Period | One journal entry per period; journal entry lines summarized | One journal entry per period; journal entry lines not summarized |

| Option               | Journal Line Type - Summary   | Journal Line Type - Detail  |
|----------------------|---|---|
| Summarize by GL Date | One journal entry per day based upon the GL date; journal entry lines summarized              | One journal entry per day based upon the GL date; journal entry lines not summarized          |
| No Summarization     | Level of detail in General Ledger is the same as the level of detail in Subledger Accounting. | Level of detail in General Ledger is the same as the level of detail in Subledger Accounting. |

### Reversal Method

Use the Reversal Method option to determine how the reversal subledger journal entries are generated in Subledger Accounting for a given application and ledger. The options are:

- Switch DR/CR (default): reverses a debit for a credit and vice-versa
- Change Sign: The reversal entry keeps the same side as the original entity, but the sign is changed.

### Use Primary Ledger Amount

If Yes is selected, Subledger Accounting calculates the amounts for the secondary ledger using the amounts from the primary ledger. If No is selected, the secondary ledger is not calculated from the primary ledger. This field is only displayed for a secondary ledger if the corresponding subledger application has been enabled for Valuation Method.

### 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.

**Note:** For examples described in the Rounding Rule Options table below, assume that the precision is 2 and the minimum account unit is 0.01.

### ***Rounding Rule Options***

| <b>Option</b> | <b>Description</b>   |
|---------------|--|
| 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 table below describes the third party merge accounting options.

### ***Third Party Merge Accounting Options***

| <b>Option</b>                                       | <b>Description</b>  |
|---|---|
| Transfer of the third party control account balance | Results in the following accounting: <ul style="list-style-type: none"><li>• Transfers the control account balance at the merge date from the old third party to the new third party</li><li>• Reverses and rebooks the existing journal entries that occurred after the merge date</li><li>• In the case of a partial merge, transfers the balance that corresponds to the transactions that are part of the partial merge</li></ul> |
| Change of the third party                           | Updates the existing journal entries by replacing the old third party and site with the new third party and site  |
| No accounting update                                | No accounting effect  |

**Note:** The reporting currency ledgers inherit the third party merge accounting option from the primary ledger.

## Accounting Program Defaults Region

The accounting program default options are used when running the Create Accounting program and are applied to the following:

- primary ledgers
- secondary ledgers for applications subject to value method subledger

### **Accounting Program Default Options**

| <b>Option</b>           | <b>Description</b>   |
|-------------------------|--|
| Accounting Program Mode | <p>Determines a default value for the parameter in the standard report submission window when running the Create Accounting program. The value determines how subledger journal entries are created.</p> <p>The options are:</p> <ul style="list-style-type: none"><li>• <b>Draft</b><p>The accounting sequence is assigned to Subledger Accounting journal entries at the time the journal entry is completed with final status. Draft entries can be included on accounting reports but cannot be transferred to General Ledger. A draft entry does not update balances and does not reserve funds.</p><p>Draft entries can be deleted and recreated with alternative information. Users can alter the associated source transaction or accounting definition. They can alter the application setups in a way that impacts how the subledger journal entry is created.</p><p>The attributes that are common to both draft and final entries are that they are balanced accounting entries; they have an accounting date and all the data that is populated for a final entry is also populated for a draft entry.</p></li><li>• <b>Final</b><p>Used to create entries that cannot be changed once they are created. Entries are transferred to General Ledger.</p></li></ul> |
| Allow Mode Override     | Determines whether users can override the value specified in the Accounting Mode parameter when running the Create Accounting program through the standard report submission window.   |
| Transfer to GL          | Predefines whether the Transfer to GL and the General Ledger Import should occur automatically when running the Create Accounting program. This value defaults to the Transfer to GL parameter in the standard report submission window when running the Create Accounting program.  |

| Option                  | Description   |
|-------------------------|---|
| Allow Transfer Override | Determines whether users can override the value specified in the Transfer to GL parameter in the previous field when running the Create Accounting program through the standard report submission window.   |
| Post in GL              | Predefines whether the GL Posting should occur automatically when running the Create Accounting program. This value defaults to the Post in GL parameter in the standard report submission window when running the Create Accounting program.   |
| Allow Post Override     | Determines whether users can override the value specified in the Post in GL parameter when running the Create Accounting program through the standard report submission window.   |
| Accounting Report Level | Determines whether the Create Accounting report is generated in Detail or Summary mode. The Detail mode displays the details for the entries successfully accounted. The value specified defaults to the Accounting Report Level parameter in the standard report submission window when running the Create Accounting program for a batch. |
| Allow Report Override   | Determines whether users can override the value specified in the Accounting Report Level parameter when running the Create Accounting program through the standard report submission window   |
| Stop at Error Limit     | Controls the execution of the Create Accounting program. Users can set the Create Accounting program to stop when a certain number of events, specified in the Error Limit field, fail to create subledger journal entries.   |
| Error Limit             | Specifies the maximum number of failed events allowed before the Create Accounting program stops creating subledger journal entries when submitting a batch of documents  |

## Event Class Options Region

The table below describes the defaults for event class options that can be overridden.

***Defaults for Event Class Options that Can Be Overridden***

---

| <b>Options</b>        | <b>Description</b>  |
|-----------------------|---|
| Journal Category      | Defined in the AMB for an event class; can be updated with any valid journal category defined in General Ledger   |
| Use Document Sequence | Determines whether the Subledger Accounting program should use document sequencing to sequence the subledger journal entries for the first event of a document if the event class for that first event is the given event class |

---

---

## Accounting Methods Builder

### Accounting Methods Builder (AMB) Overview

The AMB enables you to create and modify subledger journal line setups and application accounting definitions. These definitions define the journal entries that enable an organization to meet specific fiscal, regulatory, and analytical requirements. These definitions are then grouped into subledger accounting methods and assigned collectively to a ledger.

By using the AMB, you can define the way subledger transactions are accounted. The AMB includes the following features that are discussed in this chapter:

- Accounting options that determine different characteristics of the journal entry
- Descriptions that appear on the subledger journal header and lines which provide additional information about the journal entry

For example, a subledger journal entry created for a Payables invoice can show the supplier name and invoice number.

- Account derivation rules to construct the accounts for a subledger journal entry line

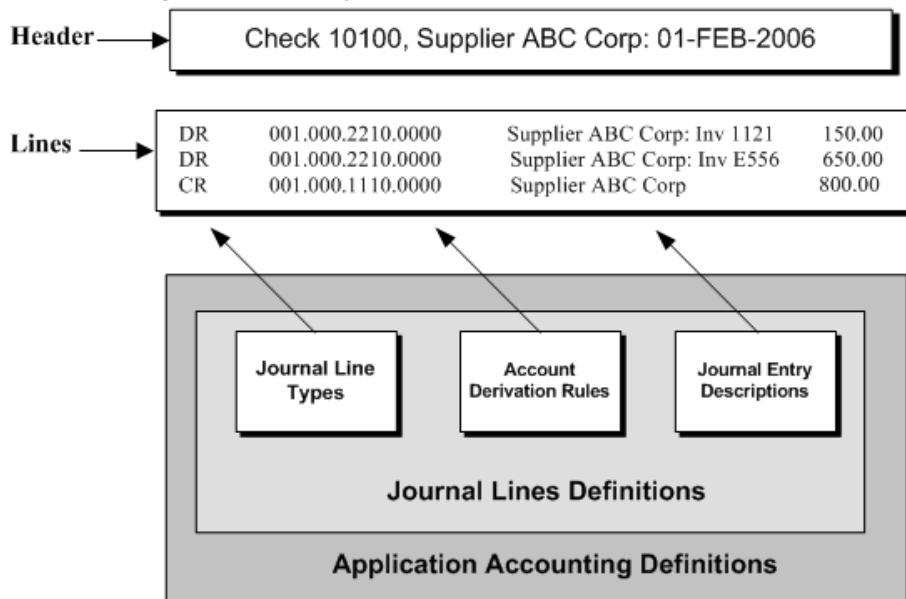
Users define various rules in the AMB to determine how a journal entry account is derived. Users can derive accounts segment by segment or as a complete Accounting Flexfield.

- Conditions that determine when subledger journal entry accounts and lines are created

Some accounts can be used to create a journal entry only in certain circumstances. For example, an asset account can only be used when the Assets flag for an invoice distribution is enabled through the Oracle Payables Invoices window.

The different elements of a subledger journal and their relationships to the AMB are displayed in the diagram below and explained in the subsequent text.

### Elements of a Subledger Journal Entry



The AMB includes journal entry setup components to configure each of these elements:

- **Journal Line Types:** Control journal entry line options such as balance type, side, and summarization
- **Journal Entry Descriptions:** Control the description for the journal entry headers and lines
- **Account Derivation Rules:** Control the derivation of Accounting Flexfield combinations for the journal entry lines

The journal entry setup components are associated with journal lines definitions that are attached to application accounting definitions. You can group detailed subledger accounting definitions for different kinds of transactions into consistent sets, each of which addresses different needs. While one application accounting definition can generate subledger journal entries to meet a particular set of requirements, another definition can be defined to satisfy completely different requirements.

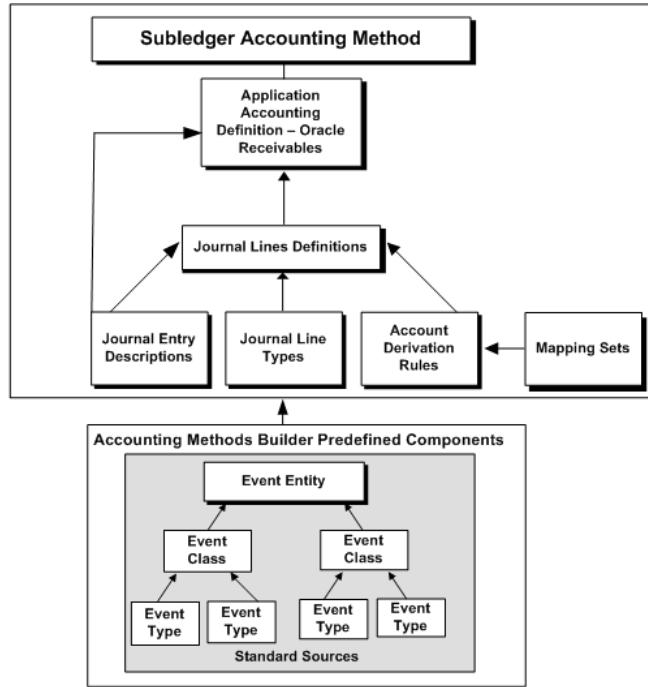
To use application accounting definitions, they must be included in a subledger accounting method and then assigned to a ledger. Users can group accounting definitions from multiple products, such as Oracle Payables, Oracle Receivables, and Oracle Assets into a single accounting method. You can assign a subledger accounting method to multiple ledgers.

As an example of these groupings and assignments, consider a set of definitions set up to create accrual accounting for payables in the U.S. These definitions can be grouped into the U.S. Payables Accrual application accounting definition. Accrual accounting application accounting definitions for each application in the U.S., such as U.S. Payables

Accrual and U.S. Receivables Accrual, can be grouped into the U.S. Accrual Subledger Accounting Method.

The relationship between the various AMB components are displayed in the diagram below and is described in the succeeding text.

### ***Accounting Methods Builder Components***



The above diagram shows how the AMB predefined components, standard sources, event entities, event classes, and event types, can be used to create journal entry descriptions, journal line types, account derivation rules, and mapping sets. The mapping sets are used in the setup of account derivation rules. The journal line types, account derivation rules, and journal entry descriptions are assigned to journal lines definitions. The journal lines definitions and optional journal entry descriptions for the journal headers, are assigned at the application accounting definition level. Application accounting definitions are grouped in a subledger accounting method.

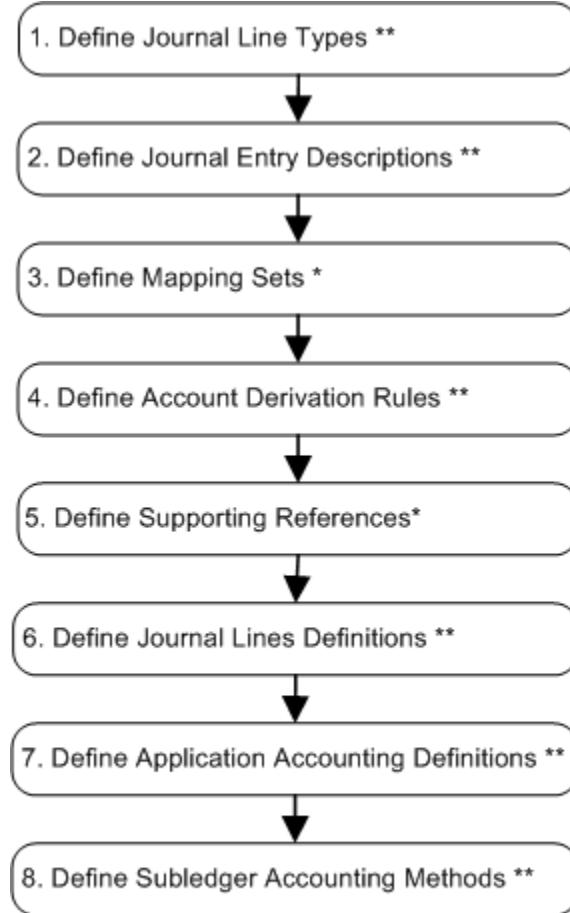
Oracle Applications development provides startup application accounting definitions and at least one subledger accounting method for all products using Oracle Subledger Accounting. If users do not have any special accounting requirements, these startup definitions may meet their needs and the only required setup step is to assign subledger accounting methods to the ledger.

If users have specific accounting requirements that are not met by the startup definitions, they can copy and modify the seeded definitions or create new definitions.

## Accounting Methods Builder Process

The Components for Building Accounting Definitions figure below shows the components in the Accounting Methods Builder process for building accounting definitions and is described in the succeeding text.

### ***Components for Building Accounting Definitions***



\* Optional Step

\*\* Start up data seeded by Oracle. Users can modify these definitions or create their own.

The components in the figure above are used by the Subledger Accounting program to create subledger journal entries. Before creating or modifying any components or definitions, check whether the seeded application accounting definitions meet your requirements. In the event that more detail is required than that provided by the default definitions, modify them or create new ones.

Since the flexibility for creating accounting definitions is dependent on source availability, Oracle subledgers provide standard sources. Create custom sources for use

in accounting definitions.

## **Step 1. Define Journal Line Types**

Set up journal line types for a particular event class and determine the characteristics of the subledger journal entry lines. Also, set up conditions for the use of the journal line type.

This is startup data seeded by Oracle. Modify these definitions or create new ones.

See: Journal Line Types, page 2-29

## **Step 2. Define Journal Entry Descriptions.**

Define the journal entry description for the subledger journal entry. Descriptions are assigned to the journal header and lines.

This is startup data seeded by Oracle. Modify these definitions or create new ones.

See: Defining Journal Entry Descriptions, page 2-47

## **Step 3. Define Mapping Sets.**

Use mapping sets in the account derivation rules setup. By mapping input values to different outputs, mapping sets provide flexibility in the creation of account derivation rules. The use of mapping sets is optional.

See: Mapping Sets, page 2-51

## **Step 4. Define Account Derivation Rules.**

Account derivation rules determine the Accounting Flexfields for subledger journal entries. You can also define conditions that determine when a particular rule is used.

This is startup data seeded by Oracle. Modify these definitions or create new ones.

See: Account Derivation Rules, page 2-57

## **Step 5. Define Supporting References.**

Supporting references may be used as follows:

- to provide additional business information about a subledger journal entry at the header or line level
- to establish a subledger balance for a particular source value or combination of source values for a particular account
- to assist with reconciliation of account balances

- for financial and managerial analysis

The use of supporting references is optional.

See: Defining Supporting References, page 2-76

## Step 6. Define Journal Lines Definitions.

Use journal lines definitions to group and assign journal line types, account derivation rules, and journal entry descriptions into a complete set of journal entries within an event class or event type. Share these sets across application accounting definitions for the same application.

The use of journal lines definitions is required.

See: Journal Lines Definitions, page 2-79

## Step 7. Define Application Accounting Definitions.

Use application accounting definitions to group journal lines definitions and header assignments for event classes and event types. You can also optionally add one or more supporting references.

Also, indicate whether to create accounting for a particular event class or event type. For example, when using cash basis accounting, you would not create a journal entry to record the accrual of an invoice.

This is startup data seeded by Oracle. Modify these definitions or create new ones.

See: Application Accounting Definitions, page 2-107

## Step 8. Define Subledger Accounting Methods.

Group application accounting definitions that comply with a common set of accounting requirements into a subledger accounting method. Each subledger accounting method can be assigned to one or more ledgers.

This is startup data seeded by Oracle. Modify these definitions or create new ones.

See: Subledger Accounting Methods, page 2-118

## Copy and Modify Functionality

If you have specific accounting requirements that are not covered by the seeded definitions, use the copy feature included in the AMB to create copies of the seeded definitions and customize them. Also, create new definitions and copy and modify them appropriately.

The copy functionality reduces the need to repeat data entry functions when there are substantial similarities between two definitions. Definitions created by users are not overwritten by upgrades to Subledger Accounting. However, upgrades can still affect

definitions if those definitions use seeded components. Use the merge analysis feature to assess whether any upgrades to the AMB impact any user-defined application accounting definitions.

The copy and modify functionality is provided for the following components of the AMB:

- Subledger accounting methods
- Application accounting definitions
- Journal lines definitions
- Journal line types
- Account derivation rules
- Journal entry descriptions
- Supporting references

Assign custom components only to custom definitions. For example, assign a custom account derivation rule (one that has been copied from an existing account derivation rule and modified) only to a custom journal lines definition. Assign a journal lines definition only to custom application accounting definitions and custom subledger accounting methods.

Use seeded components in custom application accounting definitions.

## **Transaction and Accounting Charts of Accounts**

Both the transaction chart of accounts and the accounting chart of accounts are used extensively in the AMB and are quite distinct. These fields appear in several setup windows.

### **Transaction Chart of Accounts**

The transaction chart of accounts is the chart of accounts for the primary ledger and is referenced when users enter Accounting Flexfields for their transactions. This chart of accounts is employed when users enter and maintain the data required to support the daily operations of a company.

For example, a receivables invoice is recorded in the system. The item, tax, freight, and other Accounting Flexfields that users view and enter for the invoice contain the structure and values from the transaction chart of accounts. Similarly, recording the receipt of goods and invoices as well as the issue of payments are all done in the context of a transaction chart of accounts structure.

Values for Accounting Flexfield sources stored in the transaction objects are taken from

the transaction chart of accounts. Use these source values in the AMB account derivation rules and conditions.

## Accounting Chart of Accounts

Use the accounting chart of accounts to create the Accounting Flexfields for subledger journal entries. It is taken from the ledger for which the journal entries are created.

Account derivation rules derive accounts for a specific accounting chart of accounts. The creation of all journal entries by Subledger Accounting is therefore done in the context of the accounting chart of accounts.

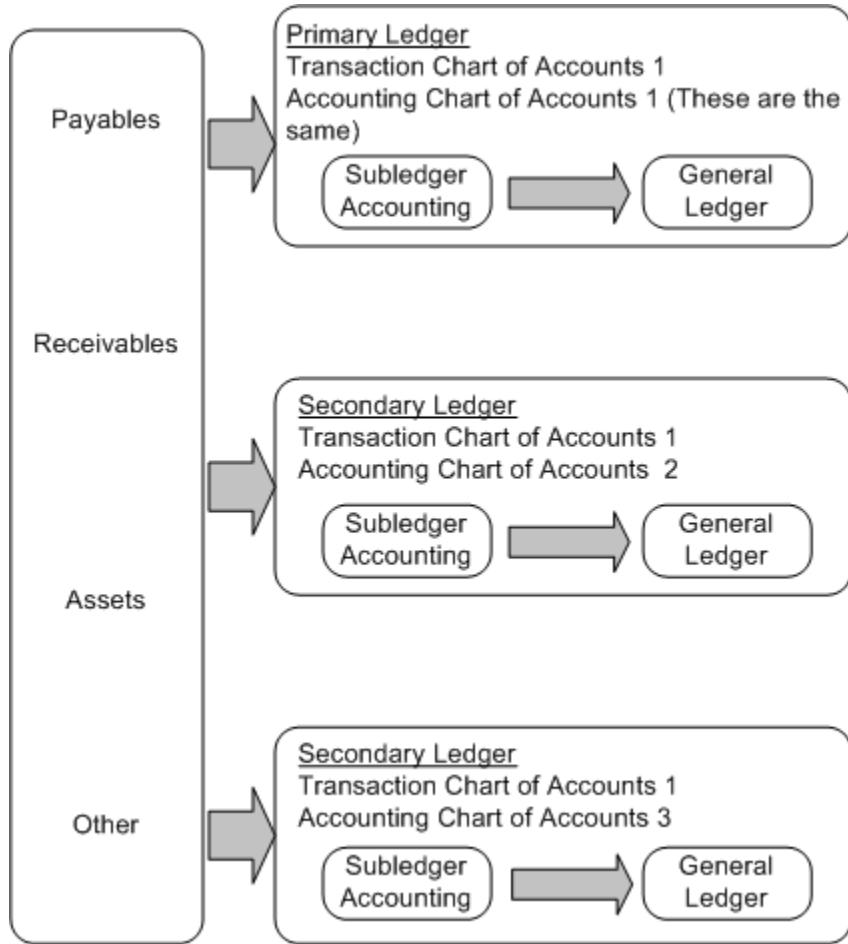
**Note:** The transaction and accounting charts of accounts are always the same for the primary ledger. They can be different in cases where users create secondary multiple representations.

## Multiple Representations

Assign each subledger accounting method to many ledgers. The combination of a subledger accounting method and ledger is called an accounting representation. The primary ledger reflects the primary accounting representation. In the primary ledger, the transaction and accounting charts of accounts are always the same.

Create multiple accounting representations by using secondary ledgers. Due to regulatory or other requirements, the accounting for these ledgers can be different than the accounting for the primary ledger. Each of these ledgers can have a different subledger accounting method assigned to it and is a secondary accounting representation. Create several secondary representations, each with a different currency, chart of accounts, calendar, and set of accounting definitions. An example of multiple accounting representations is described in the figure below.

### Multiple Accounting Representations



### Example: Multiple Representations and Transaction or Accounting Charts of Accounts

To illustrate the difference between the two charts of accounts, consider a data entry clerk in a French firm who is creating payables invoices in Payables. This individual creates transactions using the charts of accounts from the French primary ledger. The transactions are then used to create accounting events for accounting. The account code combination used during data entry is from the transaction chart of accounts. For the purposes of the illustration, this chart of accounts has a six segment structure:

- Company
- Department
- Cost Center
- Account

- Product
- Sub-Product

Assume that French local requirements include a statutorily mandated Reporting Chart of Accounts for financial reporting. In most environments, both transaction entry and accounting representation is done in the context of a single ledger. It is normal to create the accounting for this transaction using the same chart of accounts as that used for transaction entry. The Reporting Chart of Accounts is used for both transaction and accounting charts of accounts. These two charts of accounts are always the same in the primary ledger.

In addition to the Reporting Chart of Accounts, assume that the firm requires a detailed Management Chart of Accounts for analytical purposes. This Management Chart of Accounts has a four segment structure:

- Company
- Department
- Cost Center
- Account

Using the AMB, the firm can create accounting for a secondary representation in a secondary ledger, which is based on a different currency, chart of accounts, calendar, and set of accounting definitions.

In this example, it is assumed that the firm's management requires a different chart of accounts for analytical purposes. This is a secondary representation and uses the Management Chart of Accounts. When accounts and journal entries are created using the AMB, they are created for the accounting chart of accounts. The creation of all journal entries by Subledger Accounting in the secondary ledger is therefore done in the context of this Management Chart of Accounts.

## Using the Transaction and Accounting Chart of Accounts

The transaction and accounting charts of accounts are both optional for application accounting definitions and subledger accounting methods. If these charts of accounts are not used, then the AMB uses the chart of accounts assigned to the primary ledger as a default.

You can create application accounting definitions and subledger accounting methods that are completely independent of the transaction and accounting charts of accounts. Oracle uses definitions of this kind for startup data since they are created before the user has decided upon the definition of their charts of accounts.

However, some features of the AMB are dependent on the transaction or accounting chart of accounts. If the transaction chart of accounts is defined, then its individual segments are not available for use in application accounting definitions. If the

accounting chart of accounts is not defined, then you cannot set up rules to derive values for individual segments with the exception of Accounting Flexfield qualifier segments.

## Assigning the Transaction and Accounting Chart of Accounts

If any component of an account definition is set up with a particular transaction or accounting chart of accounts, then it can only be assigned to an overlying definition with the same transaction or accounting chart of accounts. For example, assign a mapping set defined for an application with a particular accounting chart of accounts only to an account derivation rule with the same accounting chart of accounts.

As another example, if an application accounting definition is defined with a particular accounting chart of accounts, then it must be assigned to a subledger accounting method with the same accounting chart of accounts. Furthermore, you can assign this subledger accounting method only to a ledger with the same matching chart of accounts.

The transaction and accounting charts of accounts cannot be different if the subledger accounting method is to be assigned to the primary ledger. For the setup of a secondary accounting representation, while the accounting chart of accounts must be equal to the chart of accounts used by the ledger, the transaction chart of accounts can be different.

## Chart of Accounts Mapping

A chart of accounts mapping is assigned at the ledger level and can be used to map Accounting Flexfield combinations from the transaction chart of accounts to the accounting chart of accounts. When subledger journal entries are generated for multiple representations, a chart of accounts mapping may be necessary to obtain values for the accounting chart of accounts in the secondary ledger.

A chart of accounts mapping is required if the following conditions are met:

- The transaction chart of accounts is different than the accounting chart of accounts.
- The account derivation rule is by Accounting Flexfield with a value type of Source.

In this scenario, the account derivation rules create accounts based on the transaction chart of accounts. In the case of the primary ledger, the derived account is the same in both the transaction and accounting charts of accounts, which are the same. However, in a secondary ledger, the account created by an account derivation rule, based in this case on the transaction chart of accounts, has a structure different than that of the secondary ledger's accounting chart of accounts. In this case, you need to map Accounting Flexfield combinations from the transaction chart of accounts to the accounting chart of accounts.

See: Account Derivation Rules, page 2-57

The use of a chart of accounts mapping imposes the following limitation: the level of detail in the transaction chart of accounts must be greater or equal to the level of detail

in the accounting chart of accounts. There should be a many to one relationship in the mapping of the transaction chart of accounts to the accounting chart of accounts. One or more Accounting Flexfield combinations from the transaction chart of accounts can be mapped to one and only one combination in the accounting chart of accounts.

## Event Model

Accounting events represent transactions that have a financial accounting impact. Examples of accounting events are issuing an invoice and disposing an asset. Financial accounting information can be recorded for these events. Accounting events cannot be compared to system events and programs that update transaction tables; instead they should be analyzed from a business perspective. Events are captured when transactions are committed in the subledgers.

As an example, a Payables invoice is created, then approved, possibly adjusted, and then paid or canceled. The accounting events representing these transactions can create one or more subledger journal entries and subsequently link the originating transaction to its corresponding journal entries.

Accounting events are categorized into event types. Event types are grouped into event classes that in turn are grouped into event entities. These groupings play a prominent role in the setup of the AMB. The definition of several components in the AMB is by event class or event type.

See:

- Accounting Methods Builder (AMB) Introduction, page 2-1

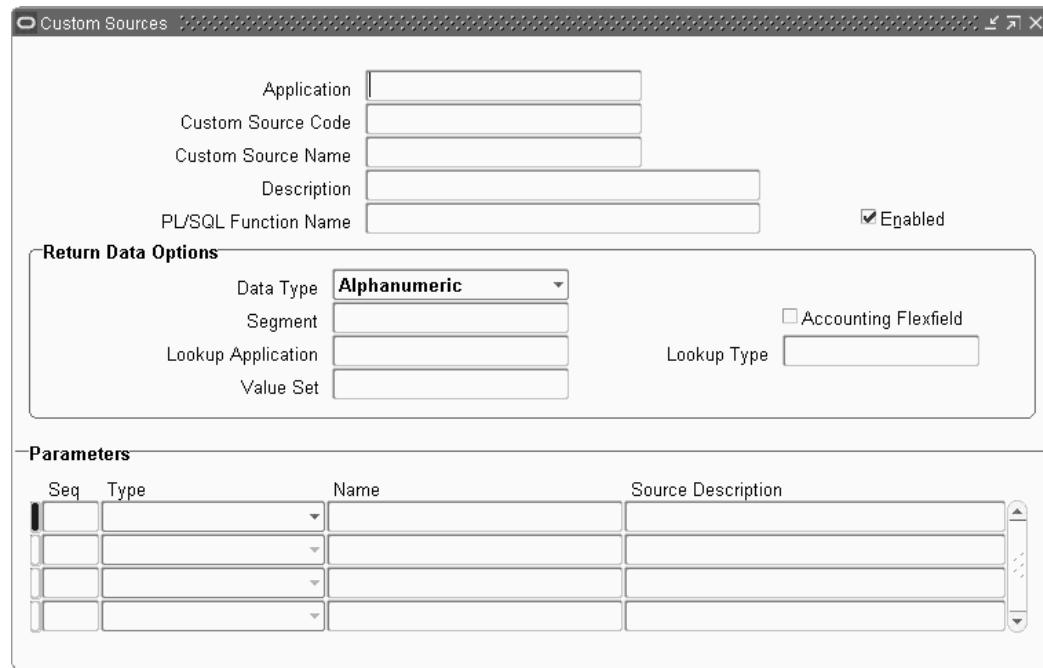
## Custom Sources

Extend the list of sources available to application accounting definitions. Using standard and system source values as parameters, write PL/SQL functions that create custom sources.

It is used to extend the list of sources available to application accounting definitions. First, decide your primary goal in defining a custom source. Also be sure to explore other alternatives, as custom source may impact the performance of the create accounting program. If you need to define one custom source, then define a PL/SQL function. In the function, you determine the input and output parameters. The output parameters becomes the return data options that you need to provide the Custom Source page. The name of the PL/SQL function must be provided in the Custom Source page. Provide the input parameters that you need in the PL/SQL function in the Parameters section of the Custom Source page. A custom source's PL/SQL function may use the seeded sources that are available to the application accounting definition or any constant value. You have to define the PL/SQL function in the database.

## Defining Custom Sources

In the Custom Sources window, define custom sources using seeded sources and constant values when sources required for the definition of accounting rules are not provided as seeded sources.



The table below describes selected fields in the Custom Sources window.

### **Selected Fields in the Custom Sources Window**

| Field                | Description   |
|----------------------|---|
| Enabled              | Retains the default to make the source available for use  |
| Segment              | Marks a source as a segment for an Accounting Flexfield   |
| Accounting Flexfield | Marks a source as an Accounting Flexfield. If selected, the Lookup Application, Lookup Type, and Value Set fields are disabled. |
| Lookup Application   | Application to associate with a lookup type. If selected, the Value Set field is disabled.                                      |

| Field     | Description   |
|-----------|---|
| Value Set | <p>The list of values includes all enabled independent and table-validated value sets and enforces the following rules:</p> <ul style="list-style-type: none"> <li>• Data type is not date.</li> <li>• The Accounting Flexfield check box is disabled.</li> <li>• No lookup application is selected.</li> </ul> |
| Seq       | Sequence in which parameters are described in the plsql function name   |
| Type      | Source type   |
| Name      | Parameter name  |

## Accounting Attributes Guideline

The accounting program uses accounting attributes values to create subledger journal entries. The types of accounting attributes values are as follows:

- 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 of this type are Entered Currency Code and Entered Amount.
- Values that control the behavior of the subledger program when processing a specific accounting event or transaction object line  
Examples of accounting attributes of this type are Accounting Reversal Indicator and Multiperiod Option.

Each accounting attribute is associated with a level:

- Header to create subledger journal entry headers
- Line to create subledger journal entry lines

## Accounting Attribute Assignments

The accounting program derives the values of accounting attributes by looking at the sources that are assigned to them. Almost all accounting attributes have sources assigned at the event class level. Depending on the accounting attribute, the accounting attribute assignment defaulted from the event class can be overridden on journal line types or application accounting definitions.

## Post-Accounting Programs

A Post-Accounting program is a container for accounting classes. Subledger applications use the Post-Accounting program assignment to determine which journal entry lines to retrieve for a particular process. For example: for mass-additions, Oracle Payables needs to define a Post-Accounting program to identify the journal entry lines that need to be pushed to Assets.

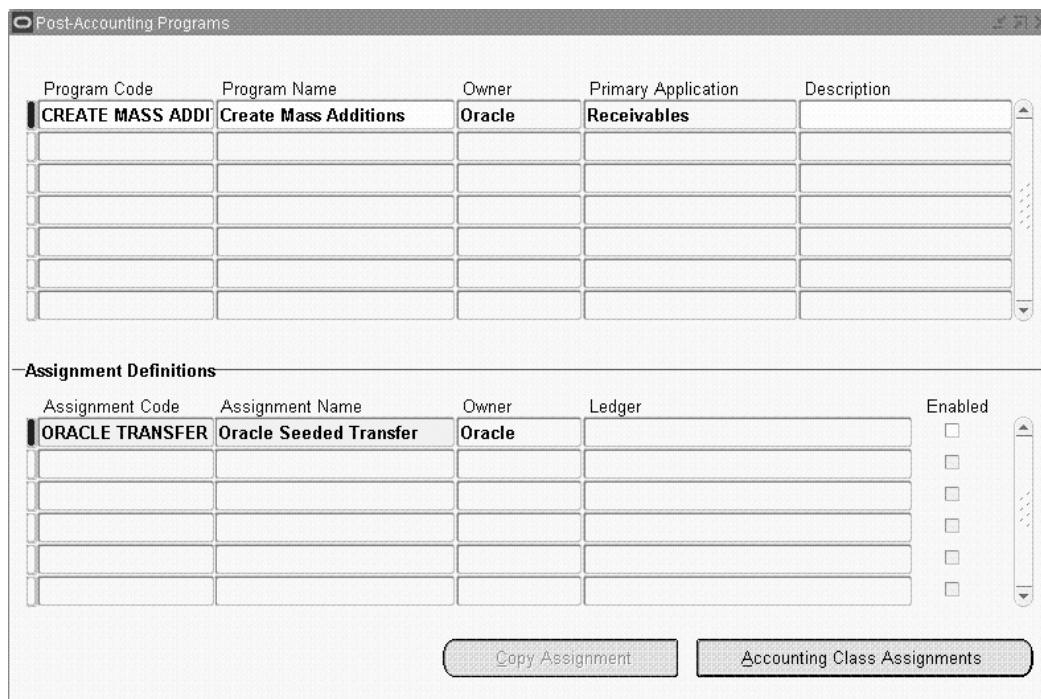
### Defining Post-Accounting Programs

In the Post-Accounting Programs window, define Post-Accounting Programs and create assignments that may be associated with a ledger.

In the Accounting Class Assignments window, assign accounting classes to an assignment definition.

Seeded programs and assignment definitions cannot be deleted or updated. You can copy a seeded assignment definition and modify the copy.

## To Define Post-Accounting Programs



The table below describes selected fields and buttons in the Post-Accounting Programs window.

### ***Selected Fields and Buttons in the Post-Accounting Programs Window***

| Field or Button     | Description   |
|---------------------|---|
| Owner               | Automatically populated. If seeded by Oracle developers, the owner is Oracle. Otherwise, the owner is User.   |
| Primary Application | Defaults from the responsibility  |
| Ledger              | Ledger to seed ledger-specific assignments; disabled for assignment definitions seeded by Oracle development  |
|                     | Enable only one assignment definition for a specific ledger. Assignments created for a specific ledger override the ledger-independent assignments. |

| Field or Button  | Description   |
|------------------|---|
| Enabled          | If selected, makes the program assignment available for use |
| Accounting Class | Opens the Accounting Class Assignments window               |

**Note:** The following fields cannot be updated once the record is saved:

- Program Code
- Assignment Code
- Ledger

## Downloading Post-Accounting Programs

You can download Post-Accounting Programs using the following syntax:

```
FNDLOAD <username>/<password>[@connect]0 Y DOWNLOAD
@XLA:patch/115/import/xlapgseed.lct <datafile> XLA_POST_ACCT_PROGS
APPLICATION_ID=<application_ID>
```

## Business Flows

Use business flows to establish a link between the accounting of transactions that are related both within the same application and across applications. With this link, you can preserve key accounting information across related transactions instead of using the same set of rules to derive this information.

The purpose of business flows is:

- Preserve General Ledger accounts or segment values across journal entries of related transactions within a business flow
- Ensure proper General Ledger balances

A business flow is a series of logically related business transactions and their accounting where the creation of one transaction causes the creation of another transaction which itself can result in another transaction.

For example, when goods are received, an invoice is entered and the invoice is paid. The business flow at the transaction level is:

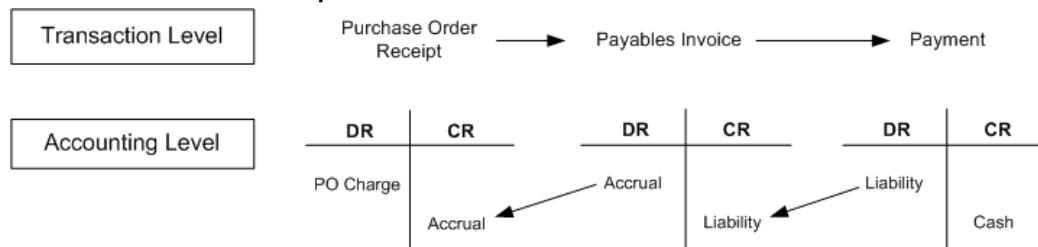
1. The receiving transaction is entered to acknowledge the arrival of the goods.
2. The invoice is entered to acknowledge that the supplier is owed for the goods and therefore references the receipt transaction.
3. The payment is created to satisfy the amount owed on the invoice and therefore references the invoice.

The business flow at the accounting level is:

1. The receipt generates an entry to the purchase order charge account.
2. This entry is offset by an accrual entry representing a future invoice liability.
3. The invoice reverses this accrual entry and creates the invoice liability, and the accounting for the payment reverses the invoice liability and generates the entry reducing the company's cash account.

The diagram below illustrates this example.

#### **Basic Business Flow Example**



**Note:** In the context of business flows, upstream refers to a transaction and its accounting that occurred before the current transaction. Downstream refers to a transaction and its accounting that come after the current transaction. For example, an invoice is upstream from the payment and a payment is downstream from the invoice.

In a business flow, it may be unreliable to copy the General Ledger account entered on an upstream transaction to the current journal entry. This is because the AMB provides the flexibility to override certain Accounting Flexfield values such that the General Ledger account on the transaction is different from the General Ledger account on its journal entry. The accounting side of a business flow is not preserved if the downstream journal entry copies the General Ledger account from the upstream transaction, which was actually accounted to a different General Ledger account.

Additionally, you can implement accounting rules that have identical conditions, but the source values used by these conditions can change between the time of the original entry and the time of the current entry. For example, a source can be a date or a project

number.

However, you can copy accounting information from the actual journal entry of a related transaction instead of evaluating the same set of conditions or copying static transaction values. In the AMB, define rules that permit a transaction's accounting to inherit certain values instead of having them determined from sources, rules, or conditions. These values are inherited from either the journal entries of a related transaction in the initial stage of the business flow or from the other side of the current journal line.

Developers provide the identifiers that link transactions in the business flow. These links are provided in the transaction objects. Subledger Accounting uses these links, along with the accounting rules, to determine the means by which accounting data is copied.

## Business Flows Features

### Business Flow Method

The business flow method specifies how a journal line obtains certain journal entry values including accounting attributes for a journal entry. A business flow method is assigned to a journal line type in the Journal Line Types window.

#### *Business Flow Method Options*

| Option | Description  |
|--------|--|
| None   | Retains the existing business rules enforced by the Journal Line Types window. This option has no impact on the accounting program and the journal line does not rely on any other journal entry. Journal entries are created using the definitions and conditions assigned to the various journal entry components. |

| Option      | Description  |
|-------------|--|
| Prior Entry | <p>Instructs the Subledger Accounting program to:</p> <ul style="list-style-type: none"> <li>• Copy the entire General Ledger account and certain accounting attribute values from the journal entry of an upstream transaction</li> <li>• Optionally copy the journal entry description into the current entry instead of assigning a separate description</li> </ul> <p>See: Copy from Prior Journal Entry Business Flow Method Process, page 2-22</p>   |
| Same Entry  | <p>Instructs the Subledger Accounting program to build the General Ledger account using segments from the offsetting entry of the current journal line. For example, it provides the option to overlay the balancing segment value on the credit side of the entry with the balancing segment value from the debit side. This option does not copy the accounting attribute values. It provides the ability to copy the journal entry description instead of assigning a separate description.</p> |

## Business Flow Class

The business flow class is a user-defined lookup that can be assigned to a journal line type and used by more than one application. The Subledger Accounting program uses the business flow class to identify the journal entry line in the initial stage of the business flow from which current journal entries copy certain journal entry values.

## Business Flow Setup Features

Set up business flow features in the following windows:

- Journal Line Types Window, page 2-21
- Journal Lines Accounting Attribute Assignments, page 2-21
- Journal Lines Definitions Window, page 2-21

### **Journal Line Types Window**

Set up the following in the Journal Line Types window:

- Assign a business flows class to a journal line type
- Select a business flows method

See: Defining Journal Line Types, page 2-31

### **Journal Lines Accounting Attribute Assignments Window**

The Journal Lines Accounting Attribute Assignments window displays all accounting attributes assigned to an event class. If the Inherit check box is enabled, the values for the selected attributes are inherited and are not available for update or entry.

### **Journal Lines Definitions Window**

Set up the following in the Journal Lines Definitions window:

- Assign a journal line type to a journal lines definition
- For Same Entry and Prior Entry, indicate that the journal lines definition inherits the journal entry description for the journal line type
- For the Same Entry and None business flows method, assign account derivation rules
- For the Same Entry and None business flows method, optionally inherit accounting segment values

See: To Define Journal Lines Definitions, page 2-81

## **Business Flows Setup Considerations and Tasks**

In setting up business flows, developers and users should perform the following steps:

1. Evaluate accounting dependencies between transactions to understand the accounting relationship between and across accounting events.

In many business flows, it is worth preserving and propagating key accounting data to all transactions within the business flow. For example, developers can use the balancing segment value of a purchase order distribution on both the invoice and payment related to the purchase order.

2. For the purpose of Prior Entry inheritance, evaluate references between related accounting events including links across applications, such as a Purchasing receipt to a Payables invoice, and within the application, such as a Receivables invoice to a Receivables cash receipt.

Applications involved in the accounting of transactions involved in the business

flow must agree on a consistent means of identifying distributions.

3. For the Same Entry Method, evaluate accounting needs within a single transaction where key values need to be consistent across both sides of the journal entry for a single transaction.
4. Ensure that a transaction distribution has a means of identifying a single transaction distribution with which it has a one-to-one relationship.

For example, a Payables invoice distribution references only one Purchasing distribution. This information is typically captured during the creation of the transaction at the final stages of the business flow and needs to be at the level of granularity consistent with the accounting of the transaction.

5. Seed necessary lookups for the business flow class.
6. Seed journal line types and journal lines definitions for business flows to act effectively.
7. Include and populate applied to columns in the transaction objects.

## Copy from the Prior Journal Entry Business Flow Method Process

The Prior Journal Entry method process is:

1. The Subledger Accounting program uses the applied to accounting attributes to identify the accounting for a transaction or event upstream in the business flow.
2. The Subledger Accounting program searches the accounting event for entries having the same business flow class as the entry currently being created.

**Note:** In budgetary control mode, the Subledger Accounting program ignores draft entries for business flows since Draft equates to funds checking which does not persist.

3. Upon finding a match, the Subledger Accounting program copies certain values from this entry to the current entry.

## Copy from Prior Entry Example

This example includes the following steps based on the purchase order described in the table below which describes two distributions for the purchase order.

**Purchase Order Distributions**

| Distribution ID | Distribution Amount | Charge Account | Accrual Account |
|-----------------|---------------------|----------------|-----------------|
| 1011            | 75.00               | 01-101-Exp     | 01-101-Accr     |
| 1012            | 25.00               | 01-102-Exp     | 01-102-Accr     |

1. A receipt for this purchase order is entered and accounted as described in the table below.

| Journal Line Types Name | Event Class    | Side | Business Flow Method | Business Flow Class |
|-------------------------|----------------|------|----------------------|---------------------|
| PO Charges              | Goods Received | DR   | None                 | NULL                |
| PO Accruals             | Goods Received | CR   | None                 | Purchased Goods     |

2. The Subledger Accounting program uses the journal line types in the table above to create journal entries as described in the table below.

**Note:** The journal line type used to create an entry is not stored on the entry and it is referenced only for clarity.

**Goods Received Accounting Event Entry**

| Line Number | PO Distribution ID | GL Account | Debit | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|--------------------|------------|-------|--------|---------------------|---------------------------------|
| 1           | 1011               | 01-101-Exp | 75.00 |        | NULL                | PO Charges                      |
| 2           | 1012               | 01-102-Exp | 25.00 |        | NULL                | PO Charges                      |

| Line Number | PO Distribution ID | GL Account  | Debit  | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|--------------------|-------------|--------|--------|---------------------|---------------------------------|
| 3           | 1011               | 01-101-Accr |        | 75.00  | Purchased Goods     | PO Accruals                     |
| 4           | 1012               | 01-102-Accr |        | 25.00  | Purchased Goods     | PO Accruals                     |
|             |                    | Totals      | 100.00 | 100.00 |                     |                                 |

The following notes relate to the table above.

- Lines 1 and 2 were created using the journal line type PO Charges, which does not use business flow functionality and has no business class.
- Lines 3 and 4 were created using the journal line type PO Accruals, which has no business flow method but does have a business flow class of Purchased Goods that is stored on the resulting lines. It is possible to have a journal line that has no business flow method but does have a business flow class.
- The GL account for all four lines was created using basic account derivation rules that copy the account from the PO distributions.

3. An invoice is entered and matched to the purchase order.

The liability assigned to the invoice is 01-000-Liab. The table below describes the invoice distributions.

#### ***Invoice Distributions***

| Distribution Number | Distribution ID | Distribution Amount | GL Account  | Applied to Distribution ID |
|---------------------|-----------------|---------------------|-------------|----------------------------|
| 1                   | 201             | 75.00               | 01-117-Chrg | 1011                       |
| 2                   | 202             | 25.00               | 01-118-Chrg | 1012                       |

4. The invoice is accounted.

The Subledger Accounting program uses the journal line types described in the

table below to create a journal entry for the Invoice Validated accounting event.

***Journal Entry for Invoice Validated Account***

| Journal Line Type Name | Event Class       | Side | Business Flow Method | Business Flow Class |
|------------------------|-------------------|------|----------------------|---------------------|
| AP Accruals            | Invoice Validated | DR   | Prior Entry          | Purchased Goods     |
| AP Inv Liability       | Invoice Validated | CR   | None                 | NULL                |

5. The accounting for the invoice creates the journal entry lines described in the table below.

***Invoice Validated Accounting Event Entry***

| Line Number | Invoice Distribution ID | GL Account           | Debit  | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|-------------------------|----------------------|--------|--------|---------------------|---------------------------------|
| 1           | 201                     | 01-101-Acc<br>r      | 75.00  |        | Purchased Goods     | AP Accruals                     |
| 2           | 202                     | 01-102-Acc<br>r      | 25.00  |        | Purchased Goods     | AP Accruals                     |
| 3           | 201                     | 01-000-Lia<br>bility |        | 75.00  | Purchased Goods     | AP Inv Liability                |
| 4           | 202                     | 01-000-Lia<br>bility |        | 25.00  | NULL                | AP Inv Liability                |
| Totals      |                         |                      | 100.00 | 100.00 |                     |                                 |

Lines 1 and 2 of the journal entry of the Invoice Validated accounting event are created as follows:

1. The Subledger Accounting program uses the identifiers in the Applied to Distribution column in the Invoice Distributions table, page 2-24, to find the most recent journal entry for those distributions.

These values are entered in the PO Distribution ID column of the Goods Received Accounting Event Entry table, page 2-23.

2. The Subledger Accounting program searches this entry for lines that have the same business flow class as the current entry.

In this example, PO distributions 1011 and 1012 each occur twice in the journal entry as described in the Goods Received Accounting Event Entry table, page 2-23.

Only lines 3 and 4 of the Goods Received accounting event have the same business flow class as the current lines, which are described in the Invoice Validated Accounting Event Entry table, page 2-25.

3. The Subledger Accounting program copies the GL account in lines 3 and 4 of the Goods Received Accounting Event Entry table, page 2-23, from these lines onto the current lines, which are described in the Invoice Validated Accounting Event Entry table, page 2-25.

Had the General Ledger account described in the Invoice Distributions table, page 2-24 from the invoice distributions been used, the accounts credited on lines 3 and 4 of the Goods Received Accounting Event table, page 2-23 would never have been reversed leading to unbalanced General Ledger accounts.

**Note:** There is no special rule used to derive the invoice liability account. The liability account on lines 3 and 4 of the Invoice Validated Accounting Event Entry table, page 2-25 are taken directly from the General Ledger account specified on the invoice itself.

## Copy from the Same Entry Business Flow Method Process

When the business flow method is Same Entry, specify which accounting segment values should be copied from one side of the current entry to the other side of the current entry. For example, the balancing segment value used on the debit side of an entry can be copied to the credit side.

The purpose of the Same Entry business flow method is to support cases where certain General Ledger account segment values must be preserved throughout the life of the business flow. An example is when the business process requires that a cost center incurring an expense must also bear the invoice liability and cash outlay.

## Copy from Same Entry Example

Consider the example described in the table below. For this entry, the General Ledger account for lines 3 and 4 was taken directly from the invoice header. However, this does not spread the liability across the cost center (101, 102) that incurred the cost, which is borne entirely by cost center 000.

| Line Number | Invoice Distribution ID | GL Account       | Debit  | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|-------------------------|------------------|--------|--------|---------------------|---------------------------------|
| 1           | 201                     | 01-101-Accr      | 75.00  |        | Purchased Goods     | AP Accruals                     |
| 2           | 202                     | 01-102-Accr      | 25.00  |        | Purchased Goods     | AP Accruals                     |
| 3           | 201                     | 01-000-Liability |        | 75.00  | Purchased Goods     | AP Inv Liability                |
| 4           | 202                     | 01-000-Liability |        | 25.00  | NULL                | AP Inv Liability                |
|             |                         | Totals           | 100.00 | 100.00 |                     |                                 |

It is not sufficient to create an account derivation rule that takes the cost center value from the invoice distributions and overlays it onto the liability account from the invoice header. To do so would result in the Invoice Validated event journal entry described in the table below.

***Invoice Validated Event Journal Entry***

| Line Number | Invoice Distribution ID | GL Account       | Debit | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|-------------------------|------------------|-------|--------|---------------------|---------------------------------|
| 1           | 201                     | 01-101-Accr      | 75.00 |        | Purchased Goods     | AP Accruals                     |
| 2           | 202                     | 01-102-Accr      | 25.00 |        | Purchased Goods     | AP Accruals                     |
| 3           | 201                     | 01-117-Liability |       | 75.00  | Purchased Goods     | AP Inv Liability                |
| 4           | 202                     | 01-118-Liability |       | 25.00  | NULL                | AP Inv Liability                |

| Line Number | Invoice Distribution ID | GL Account | Debit  | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|-------------------------|------------|--------|--------|---------------------|---------------------------------|
|             |                         | Totals     | 100.00 | 100.00 |                     |                                 |

This is incorrect because the cost center values (117 and 118) used on lines 3 and 4 are not the same as those on lines 1 and 2 (101 and 102), which results in an imbalance in all four General Ledger accounts. The values must be copied from actual journal entry lines to ensure that the correct accounts are built.

The correct accounts are shown in the GL Account column in the table below.

#### ***Invoice Validated Accounting Event Entry***

| Line Number | Invoice Distribution ID | GL Account       | Debit  | Credit | Business Flow Class | Created Using Journal Line Type |
|-------------|-------------------------|------------------|--------|--------|---------------------|---------------------------------|
| 1           | 201                     | 01-101-Accr      | 75.00  |        | Purchased Goods     | AP Accruals                     |
| 2           | 202                     | 01-102-Accr      | 25.00  |        | Purchased Goods     | AP Accruals                     |
| 3           | 201                     | 01-101-Liability |        | 75.00  | Purchased Goods     | AP Inv Liability                |
| 4           | 202                     | 01-102-Liability |        | 25.00  | NULL                | AP Inv Liability                |
|             |                         | Totals           | 100.00 | 100.00 |                     |                                 |

To accomplish this, the business flow method on the journal line type AP Inv Liability is changed from None to Same Entry. To create the General Ledger account, the journal lines definition has two rows in the Account Derivation Rules tab: one to derive an entire account (All Segments) and the other enabled to inherit the balancing segment value.

The journal entry is created as follows:

1. The Subledger Accounting program determines that the business flow method on journal line type AP INV Liability is Same Entry. For this journal line type, the Side

is Credit as described in lines 3 and 4 of the Invoice Validated Accounting Event Entry table, page 2-28.

2. After the debit side lines are created, the Subledger Accounting program returns to processing the credit lines and uses a basic account derivation rule to get the liability account from the invoice header, which is 01-000-Liab in this example.
3. The Subledger Accounting program copies the cost center segment value on the debit lines and overlays it onto the balancing segment of the credit lines as described in the GL Account column in the Invoice Validated Accounting Event Entry table, page 2-28.
4. The cost center segment values on the invoice distribution, described in the Invoice Distributions table, page 2-29, are not used, thereby maintaining the integrity of the General Ledger account balances for the appropriate cost center segment values from the receiving through the invoicing parts of the business flow.

#### ***Invoice Distributions***

| <b>Distribution Number</b> | <b>Distribution ID</b> | <b>Distribution Amount</b> | <b>GL Account</b> | <b>Applied to Distribution ID</b> |
|----------------------------|------------------------|----------------------------|-------------------|-----------------------------------|
| 1                          | 201                    | 75.00                      | 01-117-Chrg       | 1011                              |
| 2                          | 202                    | 25.00                      | 01-118-Chrg       | 1012                              |

## **Journal Line Types**

This section includes the following topics:

- Journal Line Types Features, page 2-29
- Defining Journal Line Types, page 2-31

## **Journal Line Types Features**

Journal line types are defined for a particular event class. They must then be assigned to a journal lines definition along with supporting references, account derivation rules, and journal entry descriptions.

The definition of a journal line type includes the following features:

- Subledger Accounting uses rounding class along with the transaction rounding reference to group journal lines tighter and calculate transaction rounding.

Subledger transaction rounding differences can occur when a transaction has multiple distributions.

- The journal entry line can have an actual, budget, or encumbrance balance type. For products like Oracle public sector applications that use encumbrance accounting, separate journal line types can be created for encumbrance lines.
- Journal line types specify if the journal line is to be a debit, credit, or gain/loss line. For example, when a Payables invoice is generated, the liability account should normally be credited. The journal line type 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 type must be defined to create this debit line.

The gain/loss amount is the difference in the ledger currency due to foreign currency fluctuations. 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 time that the two are accounted.

- Users can specify whether to merge matching journal entry lines.
- Users can specify whether the gain or loss has been calculated in the primary ledger so that the gain or loss amount is not converted to the reporting currency or non-valuation method secondary ledgers.
- Users can specify whether to derive journal entry components from a related journal entry.

The business flow method determines if and how a journal line should inherit journal entry values.

See: Business Flow Method, page 2-19 and To Define Journal Line Types, page 2-31

- Users can specify whether to apply multiperiod accounting to a journal line type. See: Multiperiod Accounting, page 2-90
- Journal entry lines are transferred to General Ledger in summary or detail mode.
- Users can define conditions to restrict the use of a journal line type by controlling when a particular journal line type is used by the Subledger Accounting program.
- See: Defining Conditions for Journal Line Types, page 2-41
- Users can assign accounting attributes.

If an accounting attribute is relevant to the accounting event's underlying transaction, then its value is required to generate subledger journal entries for the

transaction. By assigning standard sources to accounting attributes in the Accounting Attributes window, you can use standard source values as the values for accounting attributes.

See: Accounting Attributes Guideline, page 2-14 and To Define Journal Line Types, page 2-31

## Defining Journal Line Types

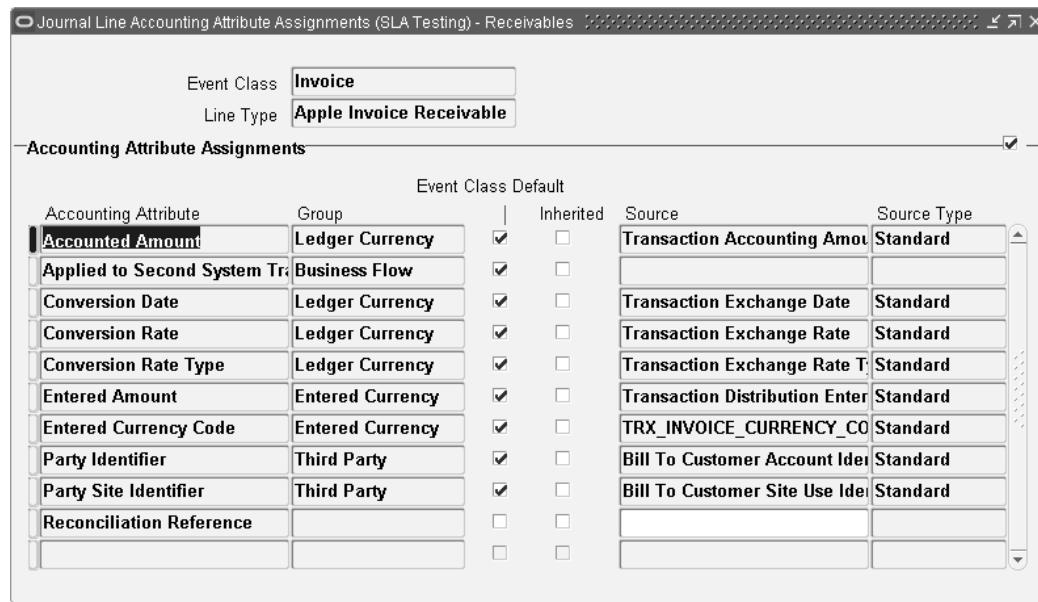
This section includes the following sections:

- To Define Journal Line Types, page 2-31
- Defining Conditions for Journal Line Types, page 2-41
- Copying Journal Line Types, page 2-47

### To Define Journal Line Types

Journal Line Types

|  |   |   |   |
|--|---|---|---|
| Application                              | SLA San Francisco Bay Area                    | Owner   | User                                    |
| Event Class                              | Bonds   | Accounting Class  | Accrual                                 |
| Line Type Code                           | ACC   | Rounding Class  | Accrual                                 |
| Name                                     | acc c   | <input checked="" type="checkbox"/> Enabled                     |   |
| Description                              |   |   |   |
| <b>Options</b>                           |   |   |   |
| Balance Type                             | <input checked="" type="radio"/> Actual       | Side  | <input type="radio"/> Debit             |
|  | <input type="radio"/> Budget                  |   | <input checked="" type="radio"/> Credit |
|  | <input type="radio"/> Encumbrance             |   | <input type="radio"/> Gain/Loss         |
| Switch Debit/Credit                      | <input type="button" value="Debit"/>          |   |   |
| Merge Matching Lines                     | No  |   |   |
| Subledger Gain/Loss                      | <input type="button" value="No"/>             |   |   |
| Chart of Accounts                        |   |   |   |
| Transaction                              | <input type="text"/>                          |   |   |
| Business Flow                            |   |   |   |
| Method                                   | <input type="button" value="None"/>           |   |   |
| Class                                    | <input type="text"/>                          |   |   |
| Multiperiod                              |   |   |   |
| <input type="radio"/> None               |   |   |   |
| <input checked="" type="radio"/> Accrual |   |   |   |
| <input type="radio"/> Recognition        |   |   |   |
| Transfer to GL                           |   |   |   |
| <input checked="" type="radio"/> Summary |   |   |   |
| <input type="radio"/> Detail (J)         |   |   |   |
| Condition <input type="text"/>           |   |   |   |
| <input type="button" value="Copy"/>      | <input type="button" value="Conditions (K)"/> | <input type="button" value="Accounting Attribute Assignments"/> |   |



The table below describes selected fields and buttons in the Journal Line Types window.

#### ***Selected Fields and Buttons in the Journal Line Types Window***

| Field, Region, and Button | Description   |
|---------------------------|---|
| Application               | Automatically populated with the application name associated with the user's responsibility.  |
| Line Type Code            | <p><b>Note:</b> Users cannot modify a seeded journal line type or any other seeded component as it could get overwritten in an upgrade. Instead users can copy the seeded type and then modify it appropriately. The copied journal line type has an Owner type of User.</p> <p>The list of values displays the component name and the owner to distinguish between seeded and user-defined components.</p> |
| Name                      | Appears in the list of values when assigning the journal line type to a journal lines definition.   |

---

| Field, Region, and Button | Description  |
|---------------------------|--|
| Accounting Class          | <p>Shared across applications and enables users to classify journal entry lines.</p> <p>For example, when a receipt is matched to a purchase order and the accrual method is On Receipt, an accrual journal line is created upon receipt creation. When the Payables invoice is matched to a purchasing document, Payables creates a journal line reversing the original accrual. In this case, Purchasing and Payables define journal line types to generate these accruals and both of them can assign the accounting class Accrual.</p> |
| Rounding Class            | <p>Accounting classes are defined using an extensible AOL lookup type. The list of values for this field contains all accounting classes that are seeded but users can add new accounting classes.</p>   |
| Enabled                   | <p>Defaults to the accounting class.</p> <p>The rounding class, along with the transaction rounding reference accounting attribute, groups lines together in order to determine whether rounding is necessary.</p>   |
| Encumbrance               | <p>If selected, makes this journal line type available for use.</p> <p>If this balance type is selected and the business flow method is not Prior Entry, an encumbrance type must be selected from the drop-down list.</p>   |
|                           | <p>See: Defining Encumbrance Types, <i>Oracle General Ledger User Guide</i></p>  |
|                           | <p>This field is disabled if Prior Entry is selected as the business flow method or if Accrual or Recognition is selected as the Multiperiod option.</p>   |

---

---

| Field, Region, and Button | Description   |
|---------------------------|---|
| Side                      | <p>A Gain/Loss journal line type creates a debit line for a loss and a credit line for a gain. The gain/loss side journal line type is used exclusively for the gain or loss amount automatically calculated by Subledger Accounting and can only be defined for the actual balance type.</p> <p><b>Note:</b> For journal line types with a side of Gain/Loss, the following accounting attributes are not displayed in the Accounting Attributes Assignments window when accessed from this window:</p> <ul style="list-style-type: none"> <li>• Applied to Application ID</li> <li>• Applied to Distribution Type</li> <li>• Applied to Entity Code</li> <li>• Applied to First Distribution Identifier</li> <li>• Applied to First System Transaction Identifier</li> <li>• Multiperiod End Date</li> <li>• Multiperiod Option</li> <li>• Multiperiod Start Date</li> <li>• Conversion Date</li> <li>• Conversion Rate</li> <li>• Conversion Rate Type</li> <li>• Entered Amount</li> <li>• Entered Currency Code</li> </ul> |

---

**Note:** If the business flow method Prior

| Field, Region, and Button | Description  |
|---------------------------|--|
|                           | <p>Entry or Same Entry is selected, the Gain/Loss option cannot be selected.</p> <p>The Gain/Loss option can only be selected if the business flow method is set to None and the Multiperiod option is also set to None.</p>   |
| Switch Debit/Credit       | <p>Determines whether negative amounts will result in negative amounts on the same side or positive amounts on the opposite side.</p> <p><b>Note:</b> If the Side is Gain/Loss, the Switch Debit/Credit field is disabled.</p> |

---

| Field, Region, and Button | Description  |
|---------------------------|--|
| Merge Matching Lines      | <p>Summarizes subledger journal lines within each subledger entry. Journal entry lines with matching criteria are merged.</p> <p>The possible values are:</p> <ul style="list-style-type: none"> <li>• All: All matching lines within a subledger journal entry with the same values for the following attributes: <ul style="list-style-type: none"> <li>• Accounting Class</li> <li>• Rounding Class</li> <li>• Transaction Rounding Reference</li> <li>• Switch Side flag</li> <li>• Gain or Loss flag</li> <li>• Business Flow Class code</li> <li>• Multiperiod Option</li> <li>• Currency</li> <li>• Conversion Rate Type</li> <li>• Conversion Date</li> <li>• Conversion Rate</li> <li>• Third Party</li> <li>• Third Party Site</li> <li>• Third Party Type</li> <li>• Accounting Flexfield</li> <li>• Description</li> <li>• Reconciliation Reference</li> </ul> </li> </ul> |

---

| Field, Region, and Button | Description   |
|---------------------------|---|
|                           | <ul style="list-style-type: none"> <li>• Gain/Loss Reference</li> <li>• Encumbrance Type</li> </ul>   |
|                           | <p>As a result of merging, a journal entry line with a net of zero amount can be created.</p> <p>Note that choosing All only merges all lines within a given subledger journal entry. This does not impact the transfer to General Ledger of summarized data. The latter is decided by making a choice in the Transfer to General Ledger region.</p>  |
| Subledger Gain or Loss    | <ul style="list-style-type: none"> <li>• No: Matching lines are not merged.</li> <li>• Dr/Cr: Matching lines with the same debit or credit side are merged to produce a single debit or credit line when the attributes listed above are matching across debit or credit lines.</li> </ul> <p><b>Note:</b> Normally, users merge matching lines. However there are some exceptions. For example, in Italy, gain or loss amounts must be recorded for each invoice payment rather than at the total payment level. By setting the merge selection to No, users guarantee that journal entry lines are not merged, even though they are for the same transaction and entry.</p> |
|                           | <p>Yes indicates that gain/loss is calculated in the primary ledger. The gain/loss amount is therefore not converted to the reporting currency and non-valuation method secondary ledgers. Select No for Subledger Accounting to calculate the gain or loss.</p> <p><b>Note:</b> When Gain or Loss is set to Yes, multiperiod accounting is disabled.</p>   |

---

| <b>Field, Region, and Button</b> | <b>Description</b>   |
|----------------------------------|--|
| Transaction                      | <p>Transaction chart of accounts.</p> <p>See: Transaction and Accounting Charts of Accounts, page 2-7 and Application Accounting Definitions, page 2-107</p>   |
| Method                           | <p>Business flow method</p> <p>If Prior Entry or Same Entry is selected, the Gain/Loss option in the Side region cannot be selected.</p> <p>If Prior Entry is selected, then the following accounting attributes are inherited from an upstream journal entry and cannot be selected or updated in the Accounting Attributes Assignments window:</p> <ul style="list-style-type: none"> <li>• Currency Code</li> <li>• Conversion Rate Type</li> <li>• Conversion Date</li> <li>• Conversion Rate</li> <li>• Party Type</li> <li>• Party Identifier</li> <li>• Party Site Identifier</li> <li>• Encumbrance Types</li> </ul> |
| Class                            | <p>Business flow class; required if the business flow method is Prior Entry</p>  |

---

---

| Field, Region, and Button | Description  |
|---------------------------|--|
| Multiperiod               | <ul style="list-style-type: none"> <li>None: Journal line type will not create multiperiod accounting.</li> <li>Accrual: To create the accrual journal line for the originating entry, such as prepaid expense</li> <li>Recognition: To create the recognition journal line</li> </ul> <p><b>Note:</b> When Gain or Loss is set to Yes, multiperiod accounting is disabled and defaults to None.</p> |
|                           | See: Multiperiod Accounting, page 2-90   |
| Transfer to GL            | <p>Select Detail to maintain the same level of detail as the subledger journal entry line. Select Summary to summarize subledger journal entry lines by Accounting Flexfield.</p> <p>One journal line is created in General Ledger to record the subledger activity.</p> <p>See: Subledger Accounting Setup Options Description, page 1-3</p>  |
| Conditions                | <p>Opens the Journal Line Type Conditions window.</p> <p>See Defining Conditions for , page 2-41 and To Define Journal Line Type Conditions, page 2-43</p>   |

---

| Field, Region, and Button        | Description   |
|----------------------------------|---|
| Accounting Attribute Assignments | <p><b>Note:</b> This button is enabled when the conditions are entered.</p> <p>Opens the Journal Line Accounting Attribute Assignments window.</p> <p>When creating a journal line type, accounting attribute assignments are automatically established based on the default accounting attribute assignments for that journal line type's event class or entity. In the Journal Line Accounting Attribute Assignments window, override this default mapping of standard sources to accounting attributes. The list of values for the Source field contains all header level sources that are assigned by developers to the accounting attribute and event class associated with the journal line type. Users can assign a source to the Reconciliation Reference accounting attribute which is used to meet accounting requirements in continental Europe.</p> |

#### ***Selected Fields in the Journal Line Accounting Attribute Assignments Window***

| Field     | Description  |
|-----------|--|
| Inherited | If selected, indicates the values of a selected accounting attribute can be inherited.   |
| Source    | <p>List of values includes all header level sources assigned to the accounting attribute and event class associated with the journal line type.</p> <p><b>Note:</b> You can assign a source to the Reconciliation Reference accounting attribute, which is used to meet accounting requirements in continental Europe.</p> |

## Defining Conditions for Journal Line Types

To set conditions appropriately, specify the journal line types that the Subledger Accounting program uses to create a subledger journal entry.

For example, set up a condition to create journal entries using a particular journal line type only if the distribution line has the Oracle Assets' tracking option set to No. As another example, an inventory transaction to record transfers could be accounted using journal line types for material overhead.

Use sources to create these conditions. For example, the condition for use of a Payables Invoice Tax journal line type could be Where line type = Tax. Similarly, the condition for a Receivables Invoice Tax journal line type could be Where account class = Tax. The line type and account class mentioned here are examples of sources.

Journal line type conditions establish whether a journal line type and its associated account derivation rules and journal entry descriptions are to be used in the subledger journal entry. Use account derivation rules to specify how an account is constructed. Associated account derivation rule conditions define the conditions under which these accounts are built.

See: Account Derivation Rules Conditions, page 2-70

## Examples of Journal Line Type Conditions

This section describes the following examples:

- Example 1: Using the Value of a Source to Set up a Journal Line Type Condition, page 2-41
- Example 2: Using Multiple Conditions with the And/Or Operator, page 2-42

### **Example 1: Using the Values of a Source to Set Up a Journal Line Type Condition**

This example describes how to create a journal line type to account for an invoice price variance (IPV). Due to changes in price, Payables uses the IPV to account for the difference between purchase order price and invoice price.

Consider an IPV journal line type. A condition is defined and attached to this journal line type, to ensure that the IPV is applied only to IPV lines. This condition can be expressed as follows:

Where

Distribution Type of the detailed distribution line = IPV.

**Using the Value of a Source to Set Up a Journal Line Type Condition Example**

| Seq. | ( | Source            | Segment | Operator | Value Type | Value | Segment | ) | And/Or |
|------|---|-------------------|---------|----------|------------|-------|---------|---|--------|
| 1    | ( | Distribution Type |         | =        | Constant   | IPV   |         | ) |        |

Now consider a case where an invoice is entered and approved. The invoice involves variance distribution lines that need to be accounted. Assume the presence of a distribution line with a Distribution Type of IPV and another with a line type of tax.

**Using the Value of a Source to Set Up a Journal Line Type Condition Example: Distribution Type and Distribution Lines**

| Source            | Distribution Line 1 | Distribution Line 2 |
|-------------------|---------------------|---------------------|
| Distribution Type | IPV                 | Tax                 |

**Using the Value of a Source to Set Up a Journal Line Type Condition Example: Journal Line Type**

| Distribution Line   | Journal Entry Line  |
|---------------------|---|
| Distribution Line 1 | IPV journal line type is used to create a subledger journal entry line. |
| Distribution Line 2 | IPV journal line type is not applied.                                   |

For distribution line 1, a journal entry line for the IPV is created based upon the journal line type. Note that a separate journal line type and condition must be written to handle the non-IPV line type.

**Example 2: Using Multiple Conditions with The And/Or Operator**

This example involves multiple conditions that can be set up with the use of the And/Or operator. Consider requirements as follows:

For cost center #420, the gain/loss on retirements from the sale of assets must use a specific journal line type.

This condition can be expressed as follows:

Where

Cost Center of the asset retired = 420 And Retirement Type = Sale

Based on the above, the account derivation rule is entered as follows:

### **Using Multiple Conditions with the And/Or Operator Example: Account Derivation Rule**

| Seq. | ( | Source               | Segment     | Operator | Value Type | Value | Segment | ) | And/Or |
|------|---|----------------------|-------------|----------|------------|-------|---------|---|--------|
| 1    | ( | Distribution Account | Cost Center | =        | Constant   | 420   |         | ) | And    |
| 2    | ( | Retirement Type      |             | =        | Constant   | Sale  |         | ) |        |

## To Define Journal Line Type Conditions

1. In the Journal Line Types window, click **Conditions**.

2. In the Seq field, enter a Sequence number.

The details of the condition are created from left to right according to this number. This number does not necessarily correspond to the evaluation of the condition, as parenthesis and logical operators precedence can affect the order.

3. In the ( field, use the "(" and ")" symbols for grouping a section of a condition.

This is useful when a condition spans multiple lines with the And/Or operators.

4. In the Source field, select a source value.

Source values are seeded by the subledger application or custom application but access to the custom sources defined is also provided. This value is the first operand in the condition line.

5. In the Segment field, if the source is a key flexfield or an Accounting Flexfield, optionally select the segment.

The table below describes the relationship between the Source and the Value Type fields based on the transaction chart of accounts.

| Transaction Chart of Accounts | Nature of Source     | Segments in List of Values   |
|-------------------------------|----------------------|--|
| Null                          | Accounting Flexfield | All Accounting Flexfield qualifiers  |
| Not Null                      | Accounting Flexfield | All segments for the transaction chart of accounts                             |
| N/A                           | Key Flexfield        | All segments for the key flexfield and flexfield structure given by the source |

6. In the Operator field, select an operator to evaluate the condition.

7. In the Value Type field, select the value type of the second operand that is evaluated in the condition line.

The value types are:

- Source: This includes any source defined for the application. The source can be seeded by Oracle or it can be a custom source.
- Constant: Choose this value type for a comparison with constant values.

The table below describes the relationship between the source and the value type fields based on the transaction chart of accounts.

| Transaction Chart of Accounts | Nature of Source     | Value Type         |
|-------------------------------|----------------------|--------------------|
| Null                          | Accounting Flexfield | Can only be Source |

| <b>Transaction Chart of Accounts</b> | <b>Nature of Source</b>           | <b>Value Type</b>         |
|--------------------------------------|-----------------------------------|---------------------------|
| Not Null                             | Accounting Flexfield              | Can be Source or Constant |
| N/A                                  | Key Flexfield or any other source | Can be Source or Constant |

When defining a condition, you can compare one data source with another data source by using the comparison operator, for example, '=' or '<>'. In some cases, the value of one side of the data source could be null or blank when the extract data contains no value for the specific source that is being compared. In such cases, Subledger Accounting will return 'TRUE' or 'FALSE' depending on which comparison operators are used and which data source is being compared as shown in the table below.

| <b>Condition</b>                             | <b>Expected Value</b> |
|--|-----------------------|
| null source = constant or a non-null source  | false                 |
| null source <> constant or a non-null source | true                  |
| null source = null source                    | true                  |
| null source <> null source                   | false                 |

8. If the Segment field to the left of the Independent Value field has a dependent segment and the value type is Constant, in the Independent Value field, enter the value of the segment that the dependent segment is based upon.

This is the value for the independent segment in the key flexfield that the dependent segment value is based upon. For example, the following condition, described in the table below, requires users to choose an independent value before choosing a value for the dependent segment.

| <b>Source</b>            | <b>Segment</b> | <b>Operator</b> | <b>Value Type</b> | <b>Independent Value</b> | <b>Value</b>   |
|--------------------------|----------------|-----------------|-------------------|--------------------------|----------------|
| Asset Category Flexfield | Minor Category | =               | Constant          | BUILDING                 | ADMINISTRATION |

9. In the Value field, enter a value as follows:

- If a value type of Constant is selected, then enter a value for the constant and this value is used as the second operand to evaluate the condition.
- If a value type of source is selected, then select an Accounting Flexfield code combination identifier or any other source value from the list of values for this field.

The table below describes the relationship between the source, segment, value type, and Value field.

| Source           | Segment       | Value Type | Value Field List of Values   |
|------------------|---------------|------------|--|
| Key Flexfield    | Not populated | Constant   | Complete key flexfield combination                                   |
| Key Flexfield    | Populated     | Constant   | Values for the segment selected                                      |
| Any other source | N/A           | Constant   | Values based on value set or lookup type if specified for the source |
| Key Flexfield    | Not Populated | Source     | All key flexfield sources  |
| Key Flexfield    | Populated     | Source     | Includes all segment and key flexfield sources                       |
| Segment          | N/A           | Source     | All segment sources  |
| Any other source | N/A           | Source     | All sources of matching data type                                    |

10. If a value type of Source and a value that represents an Accounting Flexfield code combination identifier are entered, optionally select a segment name in the Segment field.

This segment value is used as the second operand in the condition line.

11. In the `)` field, use AND/OR values to concatenate two lines together when making complex conditions.

Typically, the AND takes precedence over the OR. Parenthesis are only required to overwrite this precedence. As an example, consider a condition that should use a particular account derivation rule if the following are true:

- The Assets' tracking option is Yes AND.
- The cost is less than \$200 or the supplier is ABC.

Not only must the Assets' tracking option be Yes, but either or both of the remaining two conditions must also be true.

This condition utilizes the bracket operators as follows:

`(Asset Tracking option = Yes) AND ((Amount < 200) OR (Supplier = ABC))`

## Copying Journal Line Types and Their Associated Conditions

See Copy and Modify Functionality, page 2-6.

**Note:** If entering a transaction chart of accounts, the journal line type that is created supports this chart of accounts.

This field can only be entered if the line type being copied does not have a chart of accounts already assigned to it. If it does, the new line type that is created inherits the assigned chart of accounts and users cannot change the value.

If copying a nontransaction chart of accounts specific journal line types to a transaction chart of accounts specific journal line type, Subledger Accounting replaces any reference to Accounting Flexfield qualifiers in the conditions with the actual segment names for the qualifiers in the transaction chart of accounts. If no corresponding segment exists, an error message appears.

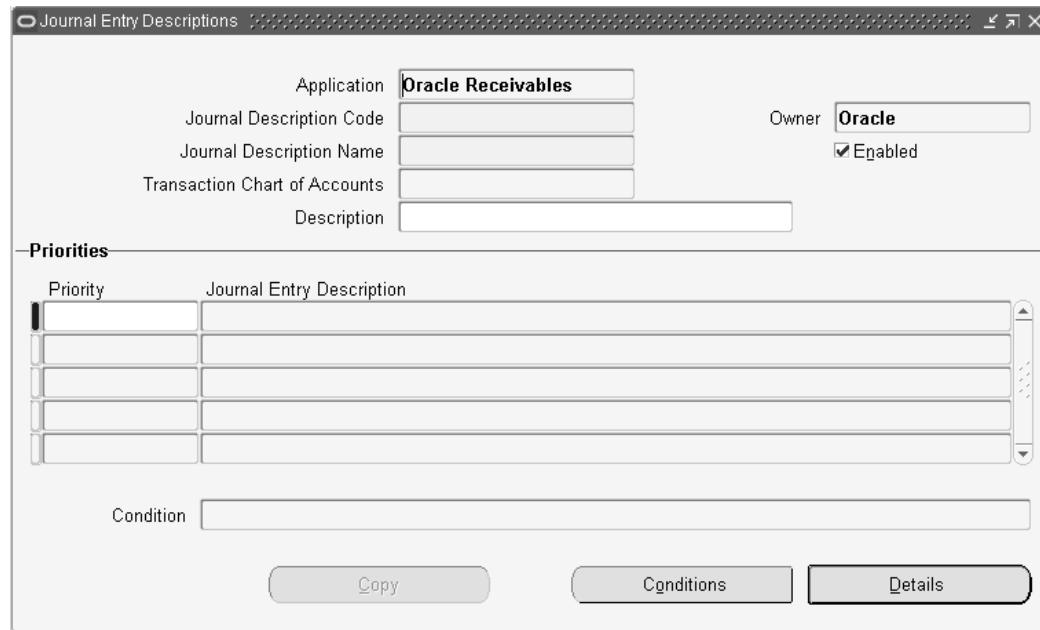
## Defining Journal Entry Descriptions

Use the Journal Entry Descriptions window to define the elements of a description that appear on the subledger journal header and line. The definition determines both the content and sequence in which the elements of the description appear.

Build descriptions using any of the available sources for the application. For example, an individual segment of an Accounting Flexfield can be included in the description. Use literal strings or a combination of both sources and literals. Some of the sources can only be used for descriptions applied at the line level. For example, a source associated with invoice line number can only be applied to a line description.

Assign journal entry descriptions headers and lines in the Applications Accounting Definition window.

## To Define a Journal Entry Description



The table below describes selected fields in the Journal Entry Descriptions window.

### ***Selected Fields in the Journal Entry Descriptions Window***

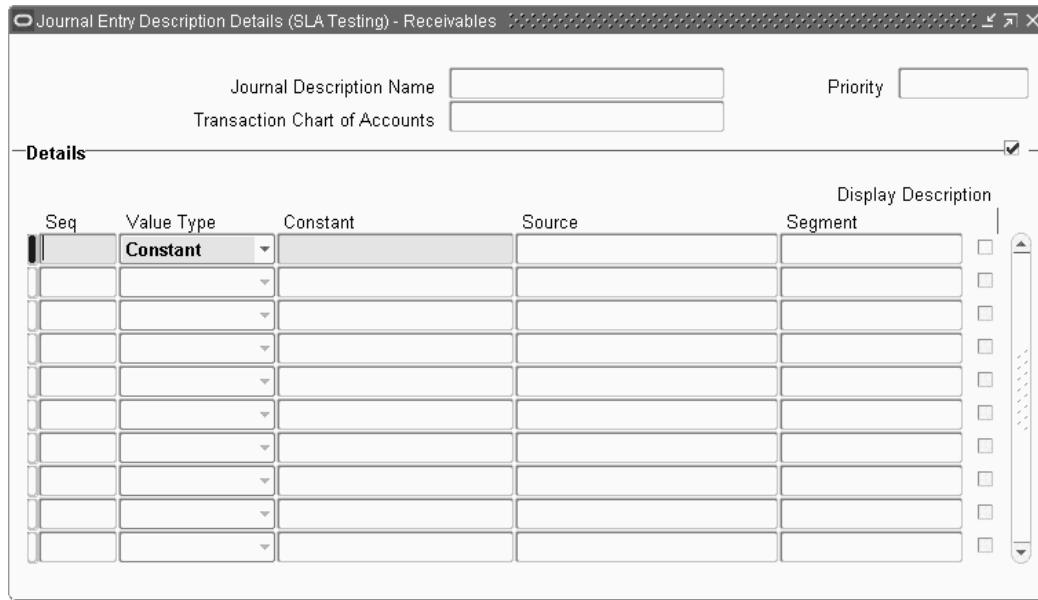
| Field                         | Description   |
|-------------------------------|---|
| Application                   | Defaults from the application associated with the responsibility  |
| Owner                         | Automatically populated. Value is Oracle for seeded components. Value is User for components created on site by users.  |
| Transaction Chart of Accounts | Chart of accounts used to input and create transactions<br>See: Transaction and Accounting Charts of Accounts, page 2-7 |

| Field    | Description  |
|----------|--|
| Enabled  | Retain the default to make the description available for assignment to application accounting definitions  |
| Priority | <p>Users can create more than one priority number for a journal entry description, each one with its own conditions and details.</p> <p>Description details are evaluated in ascending priority order, where the highest priority has the lowest number until a condition is met. For performance reasons, use the most commonly met descriptions first.</p> |
|          | <p>Change the order in which description details are evaluated by updating the priority number instead of deleting and rewriting the descriptions. Once the conditions associated with a detail line are satisfied, the value from that line is used and other lines are ignored.</p> <p>See: Journal Entry Description Conditions, page 2-51</p>            |
|          | <p>If none of the conditions are met, enter a last line with no conditions associated to it. Because Subledger Accounting uses this line as a default, assign the lowest priority to this line.</p>  |
|          | <p>The Journal Entry Description field displays a concatenation of the elements of the description created in the Journal Entry Description Details window as described in To Define Journal Entry Description Details, page 2-49.</p>   |

## To Define Journal Entry Description Details

Use the Journal Entry Description Details window to create the journal entry description that users see. Define the description content and the sequence in which it appears.

A journal entry description that uses sources that have values which vary by line cannot be assigned as a header level description.



#### **Selected Fields in the Journal Entry Description Details Window**

| Field    | Description   |
|----------|---|
| Seq.     | <p>Number for ordering the elements of a description</p> <p>Use this field to quickly change the order of appearance of the journal entry description contents.</p>   |
| Constant | <p>If Constant is selected in the Value Type field, enter text to be used in the journal entry description.</p> <p>For example, enter Check Number or Invoice Number. By defining several lines, users can use multiple sets of constant strings in the description definition. These can be concatenated with a source, as described in the Source field, and appear according to the sequence number of the line.</p> |

| Field               | Description   |
|---------------------|---|
| Source              | Note that the source selected serves as a placeholder. When the journal entry is created, this source value is replaced with appropriate source content. For example, the source Supplier Name populates the journal entry description with the actual name of the supplier. The same logic applies if the source is an Accounting Flexfield segment. |
| Segment             | Define which segment value appears in the accounting description. If a transaction chart of accounts is specified, the list of values for this field displays the name of all the segments. If there is no transaction chart of accounts specified, the list of values displays the flexfield qualifiers available in General Ledger.                 |
| Display Description | Displays the segment description for an accounting flexfield segment source.  |

## Journal Entry Description Conditions

Create journal entry description conditions in the same manner as described in Defining Conditions for Journal Line Types, page 2-41.

## Copying Journal Entry Descriptions

See: Copy and Modify Functionality, page 2-6 and Copying Journal Line Types, page 2-47.

## Mapping Sets

Use mapping sets to associate a specific output value for an Accounting Flexfield or Accounting Flexfield segment. Based on the input value, a specific value can be assigned to a single segment or to the entire Accounting Flexfield. Use mapping sets in account derivation rules to build the Accounting Flexfield.

To define a Mapping Set, pairs of values are specified. For each input value, specify a corresponding account segment or Accounting Flexfield output value. One or more related pairs of these input values and segment or Accounting Flexfield output values form a mapping set. Use value sets or lookup types for validating the input values of the mapping set.

For example, it is possible to create a mapping set based on two input values, Yes and No. Apply these input values to determine the balancing segment value of an account: 01 if the input value is Yes and 02 if it is No. Use this mapping set in one of the rules that builds the segment values of an account. The rule compares the value of a source to see if it is Yes or No and determines the segment value accordingly.

As another example, suppose a business has three major regions: East, South, and West. Assume also that the business has a Region Code segment in the Accounting Flexfield. Region names can be input values in applications such as transaction type names and service codes. These input values can be included with other information about the transaction and become part of the source information available to the AMB. Users can create a mapping set that maps region names to the corresponding region code as described in the table below.

#### ***Mapping Region Names to Region Codes***

| <b>Input Value</b> | <b>Segment Value</b> |
|--------------------|----------------------|
| East               | 01                   |
| South              | 02                   |
| West               | 03                   |

It is possible to restrict a mapping set to a range of dates by entering the start and end dates. The Event date of the potential subledger entry is compared to the effective date range of the mapping set. This determines whether mapping set values should be applied.

## **Defining Mapping Sets**

### **Prerequisite**

Using value sets and lookup types prevents data entry errors when entering input values. If planning to use either value sets or lookup types to validate the input values, they must be defined before setting up mapping sets. If neither a value set nor a lookup type is selected, users can enter any value in the Input Value field.

See: Overview of Values and Value Sets, *Oracle Applications Flexfields User Guide*

## To Define Mapping Sets

Mapping Sets (SLA Testing)

Mapping Set Code:

Mapping Set Name:

Description:

Enabled:

**Input**

Lookup Application:

Lookup Type:

Value Set:

**Output**

Accounting Chart of Accounts:

Type:

- Flexfield
- Segment
- Value Set

**Mapping Set Values**

| Value Type | Input Value | Output Value | Start Date | End Date | Enabled                             |
|------------|-------------|--------------|------------|----------|-------------------------------------|
|            |             |              |            |          | <input checked="" type="checkbox"/> |
|            |             |              |            |          | <input type="checkbox"/>            |
|            |             |              |            |          | <input type="checkbox"/>            |
|            |             |              |            |          | <input type="checkbox"/>            |
|            |             |              |            |          | <input type="checkbox"/>            |
|            |             |              |            |          | <input type="checkbox"/>            |

### Selected Fields in the Mapping Sets Window

| Field       | Description   |
|-------------|---|
| Enabled     | If selected, makes the mapping set available for use  |
| Type region | Based on this selection, the mapping set is defined for the setup of the entire Accounting Flexfield, an individual segment in the flexfield, or a value set.                           |
| Segment     | The Flexfield and Segment options are disabled if the accounting chart of accounts is not specified. The Value Set option is disabled if the accounting chart of accounts is specified. |
| Value Set   | If selected, enter the segment to be used to assign a value the mapping set.  |

| Field       | Description   |
|-------------|---|
| Value Type  | <p>If the Value Type is Input, then enter an input value.</p> <p>If Input is selected and a value set or lookup type is defined, then the list of values includes all the allowable values defined for the lookup type or value set.</p> <p>Use the value type Default in cases where the source value from the transaction is not equal to any of the input values in the mapping set. If a mapping set is used and the Subledger Accounting program encounters accounting definitions with undefined input values, journal entries cannot be successfully generated unless there is a default.</p> <p>Using the example introduced earlier in this section, if there is a source value equal to Northwest, it would not map to any of the input values. Specify a default value of 04 for any regions not included as input values. This prevents entries from failing validation when the Subledger Accounting program is submitted.</p> |
| Input Value | <p>If the value type Input is selected, enter an input value to be mapped to a corresponding output value.</p>  |

| Field                   | Description   |
|-------------------------|---|
| Output Value            | For a given input value, enter the corresponding output value. The account derivation rule uses this value to populate either the accounting flexfield or accounting flexfield segment.   |
|                         | If the Value Set option in the Type region is selected, the output value is an individual value from the value set entered in the Value Set field of the Type region, and the list of values includes all values from this value set.                 |
|                         | In the case where a mapping set is defined to populate an individual segment value, the output value is an individual segment value and the list of values includes all values for the segment entered in the Segment field.                          |
|                         | If the Accounting Flexfield option in the Type region of the Mapping Sets window is selected, then the output value is the entire Accounting Flexfield and the list of values includes all enabled combinations for the accounting chart of accounts. |
| Start Date and End Date | Effective range dates for the mapping set   |
| Enabled                 | If selected, makes the mapping set values available for use.  |

## Mapping Sets Examples

### Example 1.

When using the mapping set, the source values from the invoice includes the Small Business Option value. This is compared with the Mapping Set Input Values to determine the output segment value. Assume two transactions with the following values described in the corresponding tables.

1. In this example, assume that if procurement is from a small business supplier, then cost center 100 should be used. A mapping set is defined with two input values Yes and No as described in the following table.

***Example 1 Mapping Set***

| <b>Input Value</b> | <b>Output Value</b> |
|--------------------|---------------------|
| Yes                | 100                 |
| No                 | 200                 |

The first invoice is for the purchase of goods and services from a small business vendor. Therefore, the input value is Yes.

2. The supplier of the second invoice does not meet the criteria to be categorized as a small business. Therefore the Input Value is No.

The output segment value is therefore derived as described in the following table.

***Example 1 Output Segment Value***

| <b>Invoice</b> | <b>Cost Center</b> |
|----------------|--------------------|
| 1              | 100                |
| 2              | 200                |

***Example 2***

In this example, the chart of accounts is setup with four segments. A mapping set is defined with a value set for Supplier Type as described in the following table.

***Example 2 Mapping Set***

| <b>Input Value</b> | <b>Output Value</b> |
|--------------------|---------------------|
| Services           | 01-100-6120-000     |
| Consulting         | 01-400-6110-000     |

Assume that two invoices are entered into Payables, 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 from the accounting event

data is compared with the Mapping Set Input Values to determine the Accounting Flexfield. 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.

#### **Example 2 Derived Accounts**

| <b>Invoice</b> | <b>Supplier Type</b> | <b>Output Value</b>  |
|----------------|----------------------|----------------------|
| 1              | Services             | 01-100-6120-000      |
| 2              | Manufacturing        | No account generated |

**Note:** To ensure that Transaction 2 is accounted for, the account derivation rule to which this mapping set is assigned may have to be modified. If not, a separate rule can be defined to provide for it. Otherwise, it cannot be successfully accounted.

## **Account Derivation Rules**

Use account derivation rules to determine the Accounting Flexfields for subledger journal entries. In addition, specify the conditions under which these rules apply. Using these capabilities, develop complex rules for defining accounts under different circumstances to meet their specific requirements.

Define a rule by Accounting Flexfield, segment, or value set. If the rule is by Accounting Flexfield, the rule determines the entire Accounting Flexfield combination. For example, an account derivation rule defined by Accounting Flexfield can be used to determine the complete supplier liability Accounting Flexfield in Payables.

Define segment rules to derive the Accounting Flexfield segment by segment. For example, a particular segment like the company segment can be determined from the Distribution Accounting Flexfield. Another segment can be determined with the use of a constant value. Creating the Accounting Flexfield one segment at a time offers greater flexibility but also requires more setup.

You can use both segment-based and flexfield-based rules to derive a single account. Subledger Accounting uses segment-specific rules where they are defined and takes the remaining values from a flexfield-based rule. For example, a user can select an account derivation rule which is for All Segments and also separately select a rule which is for one particular segment. Subledger Accounting derives accounts based upon segment-specific rules and then fills in the remaining segments with the rule that specifies All Segments. Segment-specific rules take precedence over the All Segments

flexfield-based rule.

If the Create Accounting program returns an account that is end-dated or disabled and a substitute account is defined in General Ledger, Subledger Accounting uses the substitute account. The original account is stored on the journal line for audit purposes. If the substitute account is invalid and a suspense account is defined, Subledger Accounting uses the suspense account. An error message is displayed if a valid suspense account is not available.

See: Adding or Changing Individual Accounts, *Oracle General Ledger User Guide*

Define account derivation rules based on value sets in the absence of an accounting chart of accounts.

See: Overview of Values and Values Sets, *Oracle Applications Flexfield Guide*

Share account derivation rules across applications in the following ways:

- Assign an account derivation rule from the same or a different application to a journal line type in the Journal Lines Definitions window  
For example, to derive an expense account for journal line type Expense, assign the Projects Cost Account derivation rule owned by Projects to the Payables journal line type Expense.
- Create an account derivation rule based on another account derivation rule from another application and assign it to a journal line type  
For example, create a new account derivation rule Invoice Expense Account referencing Project Cost Account assigned in the Priorities region. Attach Invoice Expense Account rule to the journal line type Expense in the Journal Lines Definitions window.

**Note:** To share an account derivation rule across applications, all sources used by the account derivation rule must be available for the event class.

When an account derivation rule is assigned to a journal line type in the Journal Lines Definition window, Subledger Accounting verifies whether all sources used by the account derivation rule are available for the journal line type event class.

## Accounting Flexfield Rules

Set up Accounting Flexfield rules based upon four possible Value Types as follows:

- Source value type

For Accounting Flexfield rules, all the code combination identifiers from the transaction chart of accounts that have been setup as sources are available to create the account. Derive the account combination by specifying a source such as the

Distribution Account in the rule. The Subledger Accounting program then obtains the account by referencing the distribution Accounting Flexfield.

The transaction or accounting chart of accounts does not need to be known when this type of rule is defined. If no chart of accounts is specified, the value derived from the source uses the default chart of accounts for the subledger application.

- Constant value type

Establish the account as a constant value. For example, the constant could be the account combination 01.000.2210.0000.000 from the accounting chart of accounts specified. This is the simplest way to derive an account.

- Mapping Set value type

Derive the account combination by referencing a mapping set. Set up a mapping set rule to determine the complete account combination from the accounting chart of accounts specified.

**Note:** An accounting chart of accounts must be specified for rules using either constants or mapping sets.

- Account Derivation Rule value type

Derive the account by referencing another account derivation rule. You cannot select account derivation rules that already have an assigned account derivation rule.

The transaction or accounting chart of accounts does not need to be specified when defining this type of rule. If the account derivation rule has an accounting chart of accounts, then all account derivation rules that are assigned must have the same or no chart of accounts.

When the value type is Account Derivation Rule, select only account derivation rules with matching output types.

## Segment Rules

Set up segment rules as follows:

- When an accounting chart of accounts is specified, create a rule to derive a specific segment from the accounting chart of accounts.
- If the accounting chart of accounts is not specified, create a rule to derive the value for an Accounting Flexfield qualifier.

Set up segment rules using the same four methods discussed for Accounting Flexfield rules. By specifying different value types, users can choose the way in which the segment value is derived.

- Source value type

For Segment rules, all sources defined for the application are available. If the source is an Accounting Flexfield, take the value from a particular segment of the transaction chart of accounts. For example, derive a segment value from the cost center segment of the liability Accounting Flexfield.

In addition, use sources that are not Accounting Flexfields. For example, take the value for a Project segment from a Project Number source.

Use sources that are marked as Accounting Flexfield qualifiers and use them to derive the value for the segment.

**Note:** When creating rules for an Accounting Flexfield qualifier, only sources that are marked as Accounting Flexfield sources or the same Accounting Flexfield qualifier segment as the qualifier specified for the rule can be selected.

- Constant value type

Establish the segment value as a constant. For example, the company segment can be set to 01.

- Mapping Set value type

Use a mapping set to determine the value of the segment. For example, using a mapping set that maps supplier types to departments, the supplier type can determine the department segment value.

**Note:** An accounting chart of accounts must be specified for rules using either constants or mapping sets.

- Account Derivation Rule value type

Derive the account by referencing another account derivation rule. You cannot select account derivation rules that already have an assigned account derivation rule.

The transaction or accounting chart of accounts does not need to be specified when defining this type of rule. If the account derivation rule has an accounting chart of accounts, then all account derivation rules that are assigned must have the same or no chart of accounts.

When the value type is Account Derivation Rule, select only account derivation rules with matching output types.

## Value Set Rules

Value set based rules can be created when an accounting chart of accounts is not

specified. This enables you to share the same rule between more than one accounting chart of accounts if the segments in these chart of accounts share the same value set. Set up Value Set based rules using the same four methods discussed in Accounting Flexfield Rules, page 2-58. By specifying different Value Types, you can choose the way in which the segment value is derived.

- Source value type

For Value set based rules, all sources that are data type Alphanumeric and are not marked as Accounting Flexfields or Accounting Flexfield qualifiers are available.

- Constant value type

Establish the segment value as a constant. This list of values is based on the value set defined for the rule.

- Mapping Set value type

Use a mapping set to determine the value of the value set based rule. In this case, the mapping set must be created for the same output value set as the value set for the account derivation rule.

- Account Derivation Rule value type

Derive the account by referencing another account derivation rule. You cannot select account derivation rules that already have an assigned account derivation rule.

The transaction or accounting chart of accounts does not need to be specified when defining this type of rule. If the account derivation rule has an accounting chart of accounts, then all account derivation rules that are assigned must have the same or no chart of accounts.

When the value type is Account Derivation Rule, select only account derivation rules with matching output types.

## Account Derivation Rules Examples

Since no conditions are used in these examples, assume there is just a single detail account derivation rule line with priority one.

## Accounting Flexfield Rules Examples

1. Using a mapping set to derive an Accounting Flexfield

Consider a mapping set Vendor Category, which assigns the following accounts based on input vendor category described in the following table.

***Accounting Flexfield Rules Example 1: Mapping Set Vendor Category***

| <b>Input Value</b> | <b>Output Value</b> |
|--------------------|---------------------|
| Manufacturing      | 01-100-2210-0000    |
| Services           | 01-200-2210-0000    |
| Consulting         | 01-300-2210-0000    |

For the given transaction, assume that the accounting event information includes a Source called Supplier Type. This serves as the Mapping Set Input Source. The account derivation rule is defined to use the mapping set as described in the following table.

***Accounting Flexfield Rules Example 1: Account Derivation Rule Mapping Set***

| <b>Priority</b> | <b>Value Type</b> | <b>Value</b>    | <b>Input Source</b> |
|-----------------|-------------------|-----------------|---------------------|
| 1               | Mapping Set       | Vendor Category | Supplier Type       |

To derive an account, the values of the Mapping Set Input Source specified in a rule are compared with the input values of the mapping set. In this example, Mapping Set Input Source values from the accounting event information include the Supplier Type source values. These are now compared with the Vendor Category input values to determine what the account should be.

Assume that the accounting event data of the transaction to which the account derivation rule is applied has a value for the Source Supplier Type of Manufacturing. The account built by the Subledger Accounting program is derived by applying the account derivation rule to the transaction object data. Using the mapping set defined, if the supplier type is Manufacturing, the account created is 01-100-2210-0000.

2. The assignment of segment qualifiers to an account derivation rule enables users to create accounts with a single consistent mechanism.

See: Journal Lines Definitions Examples, page 2-79

## **Segment Rules Examples**

1. Using a specific source segment to derive the value of a different segment in a different chart of accounts.

In this example, a segment rule is created to derive the cost center segment of the Accounting Flexfield. The structure of the charts of accounts is described in the table below.

***Segment Rules Example 1: Chart of Accounts Structure***

| <b>Chart of Accounts</b> | <b>Name</b> | <b>Structure</b>                                   |
|--------------------------|-------------|--|
| Transaction              | Italy       | Balancing-Cost<br>Center-Natural<br>Account-Region |
| Accounting               | Belgium     | Balancing-Cost<br>Center-Natural Account           |

The account derivation rule is defined to derive the cost center segment value from the Region segment of the Distribution Accounting Flexfield. Note that the account is always derived for the accounting chart of accounts.

The table below describes the account derivation rule for Example 1.

***Segment Rules Example 1: Account Derivation Rule***

| <b>Priority</b> | <b>Value Type</b> | <b>Value</b>            | <b>Segment</b> |
|-----------------|-------------------|-------------------------|----------------|
| 1               | Source            | Distribution<br>Account | Region         |

Assume that the data from the Italy transaction chart of accounts has the following values for the distribution Accounting Flexfield:

02-640-2210-1234

The segment built by the Subledger Accounting program is determined by applying the account derivation rule to the source values. According to the account derivation rule, the segment to be used to derive the cost center segment of the accounting chart of accounts is the Region segment, which has a value 1234.

Note that other segment rules must be defined to build the remaining segments of the account.

- Using a source to derive the value of a segment in a different chart of accounts.

In this example, the transaction chart of accounts is not entered. Once again, a segment rule is set up to derive the cost center segment. Assume that the charts of accounts are given as described in the following table.

**Segment Rules Example 2: Charts of Accounts**

| Chart of Accounts | Name    | Structure                                |
|-------------------|---------|--|
| Transaction       | (Blank) | (Blank)                                  |
| Accounting        | Belgium | Balancing-Cost<br>Center-Natural Account |

The account derivation rule is set up as described in the following table.

**Segment Rules Example 2: Account Derivation Rule**

| Priority | Value Type | Value                   | Segment |
|----------|------------|-------------------------|---------|
| 1        | Source     | Distribution<br>Account |         |

The segment built by the Subledger Accounting program is determined by applying the account derivation rule to the source values. A segment value from the Distribution account cannot be chosen because the transaction chart of accounts is not known but a segment qualifier can be selected. Note that the account is always derived for the accounting chart of accounts.

When the Subledger Accounting program is run, if the transaction chart of accounts is the same as the accounting chart of accounts, then the resulting segment value in the Belgium chart of accounts mirrors the cost center segment value from the Distribution Account in the transaction chart of accounts.

For secondary environments, where the transaction and accounting charts of accounts are different, the segment value is determined by the chart of accounts mapping.

See: Chart of Accounts Mapping, page 2-11

Note that other segment rules must be defined to build the remaining segments of the account.

## Defining Account Derivation Rules

**Note:** Oracle strongly recommends that users do not modify a seeded rule or any other seeded component as it could get overwritten in an upgrade. Instead, copy a seeded rule and then modify it appropriately.

The modified rule has an Owner type of User.

When users select rules and other components from a list of values in AMB windows, the name as well as the Owner of the component is displayed. This enables users to distinguish between seeded and user-defined components.

## To Define Account Derivation Rules for Accounting Flexfields

The screenshot shows the 'Account Derivation Rules (SLA Testing)' window. At the top, there are fields for Application (Receivables), Rule Code, Rule Name, Description, Owner (Oracle), and a checked Enabled checkbox. Below this is a 'Chart of Accounts' section with fields for Transaction and Accounting. To the right is an 'Output Type' section with radio buttons for Flexfield (selected), Segment, and Value Set, and two empty text boxes. The main area is titled 'Priorities' and contains a table with columns: Priority, Value Type, Value, Input Source, and Segment. A 'Condition' field is at the bottom left, and 'Mapping Set' and 'Conditions' buttons are at the bottom right.

The following procedure describes selected fields.

1. Navigate to the Account Derivation Rules window, and in the Find Account Derivation Rules window, click **New**.

The application name defaults from the application associated with the responsibility.

The Owner field is automatically populated by Subledger Accounting. For components seeded by Oracle, the value is Oracle. For components created on site by users, the value is User.

2. Retain the default for the Enabled check box which is selected to make the account derivation rule available for use for any application with reference objects used in

this account derivation rule and accounting chart of accounts.

3. In the Chart of Accounts region, select values for the charts of accounts.

If a value for the accounting chart of accounts is not selected, users can create an account derivation rule for an Accounting Flexfield qualifier, an Accounting Flexfield, or a value set.

**Note:** Account derivation rules seeded by Oracle user are independent of the accounting and transaction chart of accounts.

The table below describes the accounting chart of accounts configuration.

#### ***Accounting Chart of Accounts Configuration***

| <b>Accounting Chart of Accounts</b> | <b>Accounting Flexfield Rule</b>   | <b>Segment Rule</b>   | <b>Value Set Rule</b>                           |
|-------------------------------------|--|---|---|
| Null                                | Value type must be Source and it must be an Accounting Flexfield or account derivation rule. | Value type must be source and the nature of the source can be Accounting Flexfield or segment of an Accounting Flexfield. | Value type can be Source, Mapping, or Constant. |
| Entered                             | Value type can be Source, Mapping Set, Constant, or Account Derivation Rule.                 | Value type can be Source, Mapping Set, or Constant.   | Not allowed                                     |

See: Transaction and Accounting Charts of Accounts, page 2-7

4. In the Output Type region, select the option that the account derivation rule will be based on.
5. If the output type Segment is selected, in the Segment field, select a segment from the list of values.  
If the accounting chart of accounts is specified, the list of values includes all enabled segments for the chart of accounts. If the accounting chart of accounts is not specified, the list of values includes all segment qualifiers for the flexfield application and title.
6. If the output type Value Set is selected, in the Value Set field, select a value set from

the list of values.

7. In the Priority field, enter a priority number.

Rules are evaluated in ascending order, where the highest priority has the lowest number, until a condition is met. It is advantageous to list the most commonly met rules and conditions first. The order in which rules and conditions are evaluated by updating the priority number instead of by deleting and rewriting detail lines can be changed. Once the conditions associated with a detail line are satisfied, the value from that line is used and other lines are ignored.

See: Account Derivation Rule Conditions, page 2-70

To handle the case in which none of the conditions are met, enter a last detail line with no conditions associated with it. As AMB uses this line as a default, assign the lowest priority to this line. Once a segment has a valid value, it cannot be overwritten and AMB proceeds with the building of the next segment value.

8. To specify the method of deriving the Accounting Flexfield or segment value, in the Value Type field, select a value type.
9. In the Value field, enter the value consistent with the value type selected.

The table below summarizes the relationship between accounting chart of accounts, output type, value type, and values in the Value field.

| <b>Accounting Chart<br/>of Accounts</b> | <b>Output Type</b> | <b>Value Type</b> | <b>Value Field Values</b>   |
|---|--------------------|-------------------|---|
| Null                                    | Flexfield          | Source            | All enabled sources<br>for the application<br>marked as<br>Accounting Flexfield |

| Accounting Chart<br>of Accounts | Output Type                    | Value Type              | Value Field Values  |
|---------------------------------|--------------------------------|-------------------------|---|
| Null                            | Accounting Flexfield Qualifier | Source                  | If creating an account derivation rule for Accounting Flexfield: all enabled sources for the application marked as Accounting Flexfield plus sources marked as the Accounting Flexfield qualifier corresponding to the segment to be derived in the account derivation rule |
| Null                            | Value Set                      | Source                  | If creating an account derivation rule for Accounting Flexfield: all enabled sources for the application marked as an Accounting Flexfield qualifier or Other that have the same value set  |
| Null                            | Value Set                      | Mapping Set             | All enabled mapping sets with the same value set  |
| Null                            | Value Set                      | Constant                | List of values from the value set   |
| Null                            | Value Set                      | Account Derivation Rule | All account derivation rules with no account derivation rule assigned to it   |

| <b>Accounting Chart of Accounts</b> | <b>Output Type</b> | <b>Value Type</b>       | <b>Value Field Values</b>  |
|-------------------------------------|--------------------|-------------------------|--|
| Not Null                            | Flexfield          | Source                  | All enabled sources for the application marked as Accounting Flexfield   |
| Not Null                            | Flexfield          | Mapping Set             | All enabled mapping sets with the same accounting chart of accounts and whose output type is Flexfield   |
| Not Null                            | Flexfield          | Constant                | All enabled combination for the accounting chart of accounts   |
| Not Null                            | Flexfield          | Account Derivation Rule | All account derivation rules with no account derivation rule assigned to it are available.   |
| Not Null                            | Segment            | Source                  | If creating an account derivation rule for Accounting Flexfield: all enabled sources for the application marked as Accounting Flexfield plus sources marked as the Accounting Flexfield qualifier corresponding to the segment to be derived in the account derivation rule plus sources marked as the Other segment |

| Accounting Chart of Accounts | Output Type | Value Type              | Value Field Values   |
|------------------------------|-------------|-------------------------|--|
| Not Null                     | Segment     | Mapping Set             | All enabled mapping sets with the same accounting chart of accounts and whose output type has the same segment as the segment to be derived in the account derivation rule |
| Not Null                     | Segment     | Constant                | All segment values for the accounting chart of accounts and segment  |
| Not Null                     | Segment     | Account Derivation Rule | All account derivation rules with no account derivation rule assigned to it  |

10. If value type Mapping Set is selected, in the Input Source field, select the source name to be compared with the mapping set input value.
11. If output type Segment and value type Source are selected, in the Segment field, select the segment from which the value is to be taken.  
The Segment field can be entered if the nature of source in the Value field is Accounting Flexfield or key flexfield or the nature of the input source is Accounting Flexfield or key flexfield.

## Account Derivation Rules Conditions

In the Account Derivation Rules Conditions window, specify conditions for each rule detail line. Priorities determine the order in which account derivation rule conditions are examined. Depending on which of the defined conditions is met, a different account derivation rule detail is employed to create the account.

The Subledger Accounting program evaluates conditions based on the priority of the rule detail. If the conditions for a priority are met and the associated rule detail results in a valid account or segment value, no further priorities are evaluated. Otherwise, the

Subledger Accounting program evaluates the conditions for the next priority until a valid account or segment value is derived or there are no more conditions to evaluate.

You can combine Accounting Flexfield rules with segment rules. In this case, Subledger Accounting uses the segment value derived from the segment rule to override the corresponding segment of the Accounting Flexfield. However, if the segment rule has conditions associated with the priorities and none of the conditions are met, no override occurs and therefore the segment value is derived from the Accounting Flexfield rule.

Account derivation rules conditions are created in the same manner as described in Defining Conditions for Journal Line Types, page 2-41.

## Lookup Types

If a source is a lookup code, Subledger Accounting displays its corresponding meaning in the appropriate language when it is used in header and line descriptions or supporting references. Also, when defining conditions in the AMB, Subledger Accounting displays the translated meaning to the user and stores the untranslated lookup code. Since the untranslated lookup code is used in conditions, conditions can function independently of the language used by the ledger.

Lookup types:

- Reduce the number of source values that need to be stored in the transaction object
- Help prevent errors by displaying valid, user friendly LOV names for sources that are lookup codes
- Overcome many of the restrictions associated with translated sources

## Account Derivation Rules Conditions Examples

The following example illustrates the value of specific sources to set up an account derivation rule Condition.

Consider a rule HQ Capital Purchase as follows:

The rule is to be applied only if the distribution account cost center is the same as the liability account cost center and the Assets' 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.

#### **Account Derivation Rule Condition Example: Setup**

| Seq. | ( | Source               | Segment     | Operat or | Value Type | Value             | Seg-ment    | ) | And/Or |
|------|---|----------------------|-------------|-----------|------------|-------------------|-------------|---|--------|
| 1    | ( | Distribution Account | Cost Center | =         | Source     | Liability Account | Cost Center | ) | And    |
| 2    | ( | Asset Flag           |             | =         | Constant   | Yes               |             | ) |        |

Now assume that the accounting event data, to which the account derivation rule and therefore the condition is applied, has two rows of data to be processed with the values described in the following table.

#### **Account Derivation 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 table above, assume the cost center segment as the second segment. Based on this set up, the account derivation rule is applied to derive the account of Invoice 1 only. For the second invoice, even though the Assets' tracking option is set to Yes, the cost center for the Distribution account and Liability account are not the same. Both conditions must be met in order for the rule to apply.

## **Supporting References**

Define supporting references to store transaction information on your journal entry. You can choose to maintain balances for each account and supporting reference detail value combination.

After the supporting reference is defined, assign it to the application accounting definition on the journal line types or on the header for event class and type. The accounting program will then associate values from the transaction to the journal entry created, based on your application accounting definition. You can also enter supporting references for manual journal entries. Use the online inquiry to view the account balances for a specific ledger and supporting reference.

## Supporting References Example

Example: A user decides to capture the Project Number on the Invoice Expenses lines, and defines and assigns a supporting reference on the journal line type Invoice Expense in the line assignments for the event class Invoices. The Subledger Journal Entry Example, page 2-73 describes the subledger journal entry that is booked.

### *Subledger Journal Entry Example*

| Event Class/Event Type | Journal Line Type | Account          | Amount | Supporting Reference: Project Number |
|------------------------|-------------------|------------------|--------|--------------------------------------|
| Invoice/All            | Invoice Expense   | 01-100-4001-0000 | \$400  | 100                                  |

As a result, \$400 is added to the subledger balance for the account 01-100-4001-0000 and the project number 100. It is possible to execute an online inquiry on the subledger balances by project number for the account 01-100-4001-0000.

Users can also define supporting references without associated balances. These are mainly used as journal entry references and can be assigned at the journal entry header or line levels. In contrast, supporting references defined with balances can only be assigned at the line level.

## Supporting References Setup

The following steps describe how to set up supporting references.

1. Step 1: Create the Supporting References
2. Step 2: Assign Sources to the Supporting References Details
3. Step 3: Include the Supporting References in the Header and/or Line Assignments of the Journal Lines Definition
4. Step 4: Validate the Application Accounting Definition

**Step 1: Create the Supporting References** Supporting references facilitate account analysis, reporting, and reconciliation by enabling users to break down their subledger balances in totals and subtotals. A supporting reference comprises one or more pieces of transaction information, each of which can be used to calculate a subtotal or total of the account balance.

Users must complete the following task to create a supporting reference:

Users must decide what type of information they would like to capture on journal entries and how they would like their account balances to be subtotalized and totaled.

Example 1: users determine that they want to have subtotals by transaction types and by salesperson for a revenue account. Therefore, they set up Transaction Type and Salesperson as supporting reference details.

The account balance breakdown using this grouping order is described in the table below.

***Grouping Order Example 1***

| <b>Account #01-100-3000-0000 - Sales Revenue</b> | <b>Total</b> |
|--|--------------|
| <hr/>  |              |
| Transaction Type: Consulting Sales               |              |
| Total for Salesperson: Bill Smith                | \$4,000      |
| Total for Salesperson: Sue Davis                 | \$6,000      |
| <hr/>  |              |
| Transaction Type: Software Sales                 |              |
| Total for Salesperson: Barry Hopkins             | \$6,500      |
| Total for Salesperson: Sue Davis                 | \$1,000      |
| <hr/>  |              |

Example 2: Users determine that they want subtotals by project number, by project expenditure types and by project task number for an expense account. Therefore, they set up Project Number and Project Expenditure Types as supporting reference details.

In this example, there is no subtotal for Project 2010 and expenditure type Television Ads. This would occur because no subledger journal entry lines have been posted to account 01-100-4000-0000 for this combination of Project Number and Expenditure Type.

***Grouping Order Example 2***

| <b>Account #01-100-4000-0000 - Advertising Expense</b> | <b>Total</b> |
|--|--------------|
| <hr/>  |              |
| Project Number: 1000                                   |              |
| Total for Expenditure Type: Television Ads             | \$2,000      |
| Total for Expenditure Type: Radio Ads                  | \$6,000      |
| <hr/>  |              |

---

| Account #01-100-4000-0000 - Advertising<br>Expense | Total   |
|--|---------|
| Project Number: 2010                               |         |
| Total for Expenditure Type: Radio Ads              | \$6,000 |

---

**Step 2: Assign Sources to the Supporting References Details** After users have decided the supporting references structure, they must select the sources that provide the values for the supporting references details. Each supporting reference detail can have one or more sources associated. A single source can be assigned for each application and event class that affects the account balance. This is particularly relevant when the supporting reference is used across different applications.

Example 1: Users want to break down the balance of an asset account by asset category. The source for this value in Oracle Assets can be called Asset Category, while in Oracle Projects it can be called Project Asset Category. Therefore, both sources need to be assigned to the support reference details for the applications Assets and Projects respectively.

Example 2: Users want to create a supporting reference for an inventory account to analyze the balance by inventory item. The source storing the inventory item for a purchase order can be called Purchase Order Item, while the one storing the item for a material receipt can be called Receipt Item. Therefore, both sources need to be assigned to the supporting reference detail, for the event classes corresponding to Purchase Orders and Material Receipts respectively.

**Step 3: Include the Supporting Reference in the Header and/or Line Assignments in the Journal Lines Definition** An organization can use the same general ledger account to record activity from different applications, event types and journal line types. This is especially true when the same general ledger account is used in different parts of a business cycle. To support this requirement, Subledger Accounting provides users with the ability to assign a supporting reference to multiple event classes, event types, and journal line types. This also allows the supporting reference to be independent from the general ledger accounts so that users need not know in advance all the accounts that are used in combination with certain supporting references.

Example 1: Users want to create a supporting reference to analyze revenue from Projects Billing and Account Receivables, which share the same account. Therefore, the same supporting reference must be assigned to all combinations of event class, event type, and journal line type that affects the revenue account from either Project Billing or Account Receivables.

Example 2: Users want to create a supporting reference to analyze capital projects expenditures by asset category. Therefore, the supporting reference must be assigned to all combinations of event class, event types, and journal line type used for capital expenditures in Projects as well as to all the ones used for asset additions, transfers, and

retirements in Assets.

Example 3: Users want to create a supporting reference to analyze an inventory account by inventory item. Therefore, the supporting reference must be assigned to all combinations of event class, event type, and journal line type used for inventory purchase in Payables as well as to all the ones used for inventory issuances and cost adjustments in Oracle Inventory and Cost Management.

**Step 4: Validate the Application Accounting Definition** Once users have completed the supporting reference assignments, the application accounting definition must be validated so that the changes are taken into account.

To validate an application accounting definition, the user navigates to the Application Accounting Definitions window and clicks Validate. This submits a concurrent process that validates the application accounting definition and recreates the underlying database stored procedures. Once this concurrent process completes, the status of the application accounting definition changes to Valid. This indicates that the modified definition can now be used to generate subledger journal entries.

## Year-End Carry Forward

For supporting references for which balances are maintained, users can specify whether the balances at the end of a fiscal year are carried forward to the next fiscal year. The alternatives are as follows:

- Always: The supporting reference balances at the end of the fiscal year are always carried forward to the next fiscal year.
- Never: The supporting reference balances are reset to zero at the beginning of a new fiscal year.
- Based on Account: The ending supporting reference balances for balance sheet accounts (accounts whose account type is Asset, Liability, or Owner's Equity) are carried forward to the next fiscal year, while they are reset to zero for income statement accounts (accounts whose account type is Expense or Revenue).

## Defining Supporting References

Use the Supporting References page to search for existing supporting references or navigate to the Create Supporting References page to define a new supporting reference and assign one or more sources to the details. You can define a maximum of five supporting reference details.

To define supporting references:

1. Navigate to the Supporting References window.
2. Click Create.

3. In the Code field, enter a unique code for the supporting reference using a combination of upper case letters, digits, underscores and no spaces.
4. In the Name field, enter a name for the supporting reference.
5. In the remaining fields, optionally enter information as required.
6. Click Add detail.
7. In the Detail Code field, enter a unique detail code for the supporting reference detail.
8. In the Detail Name field, enter a name for the supporting reference detail.
9. In the Description field, optionally enter a description for the supporting reference detail.
10. To assign a source to the selected supporting reference detail, click Assign Sources. You can search for sources defined for a particular application or event class and search by source name in the Assign Sources page.
11. Enter search criteria and click Go.
12. Select one or more sources from the search results and click Apply to assign the sources to the supporting reference detail.

The Supporting References page appears confirming that the supporting reference has been updated.

## Updating Supporting References

Use the Update Supporting Reference page to modify existing supporting references.

To update supporting references:

1. Navigate to the Supporting References window.
2. Enter search criteria as required.
3. Click Go.
4. Click the Update icon next to the supporting reference you wish to update.

**Note:** The Update icon is disabled if the supporting reference is defined by Oracle, since Oracle-defined supporting references cannot be updated.

5. Make changes as required in the Update Supporting References page.

**Note:** You cannot update the Maintain Balances, Year End Carry Forward, and Stored Value options if the supporting reference is assigned to any application accounting definition.

## Duplicating Supporting References

To duplicate supporting references:

**Note:** You can duplicate the supporting reference details or both the details and the source assignments.

1. Navigate to the Supporting References window.
2. Enter search criteria as required.
3. Click Go.
4. Click the Duplicate icon next to the supporting reference you wish to duplicate.
5. In the Duplicate Supporting Reference Options page, enter the code and name of the new supporting reference.
6. Choose whether to copy the details or the details and the source assignments.

If you choose the details option, the detail code, name, and description are copied from the original supporting reference. Source assignments are not copied.

If you choose the details and source assignments option, both supporting reference details and the source assignments are copied.

7. Click Continue.
8. Assign sources as required.
9. Click Apply.

A confirmation page indicates that you have successfully copied the supporting reference.

## Deleting Supporting References

To delete supporting references:

**Note:** Supporting references cannot be deleted if they are assigned to an application accounting definition or if they have already been used on journal entries.

1. Navigate to the Supporting References window.
2. Enter search criteria as required.
3. Click Go.
4. Click the Delete icon next to the supporting reference you wish to delete.

**Note:** The Update icon is disabled if the supporting reference is defined by Oracle, since Oracle-defined supporting references may not be updated.

5. A warning message will ask if you are sure that you want to delete the supporting reference, click Yes to continue or No to cancel.

## Journal Lines Definitions

Use journal lines definitions to create sets of line assignments for an event class or event type. These sets can be shared across application accounting definitions.

In the Journal Lines Definitions window, you can:

- Assign account derivation rules, supporting references, and descriptions to journal line types
- Define multiperiod accounting rules for a journal line type

## Journal Lines Definitions Examples

### Example 1

A public sector agency buys computer equipment for \$10,000. Besides accounting for the purchase, the agency wants to set aside money to cover for its replacements. The required journal entries are described in the table below.

| Equipment             | Debit    | Credit |
|-----------------------|----------|--------|
| DR Computer Equipment | \$10,000 |        |

| Equipment                        | Debit    | Credit   |
|----------------------------------|----------|----------|
| CR Account Payable               |          | \$10,000 |
| DR Fund Balance                  | \$10,000 |          |
| CR Reserve for Capital Equipment |          | \$10,000 |

To achieve the above journal entries, two journal lines definitions are required as described in the following tables.

#### ***Standard Invoice Journal Lines***

---

##### **Journal Line Type**

---

Expense

Liability

---

#### ***Public Sector Invoice Journal Lines***

---

##### **Journal Line Type**

---

Fund Balance

Reserve for Equipment

---

## **Example 2**

This example describes how to allocate an invoice's liability amount across multiple balancing segments on the invoice distributions.

The Accounting Flexfield structure is Balancing Segment-Cost Center-Account. The default liability account for supplier site ABC is 000-000-2300. If automatic offsets using the balancing method are enabled, an invoice entered for supplier site ABC is distributed as described in the table below.

| Account                   | Debit | Credit |
|---------------------------|-------|--------|
| DR Expense 101-200-1245   | \$30  |        |
| DR Expense 201-300-3045   | \$50  |        |
| CR Liability 101-000-2300 |       | \$30   |
| CR Liability 201-000-2300 |       | \$50   |

To achieve the above journal entry, the journal line definition is defined as described in the table below.

#### ***Invoice Journal Lines with Automatic Offsets***

##### **Journal Line Type**

Expense

Liability

The Liability journal line type requires the following account derivation rules:

| Segment           | Account Derivation Rule Name           |
|-------------------|--|
| All Segments      | Liability Account                      |
| Balancing Segment | Invoice Distribution Balancing Segment |

## **Defining Journal Lines Definitions**

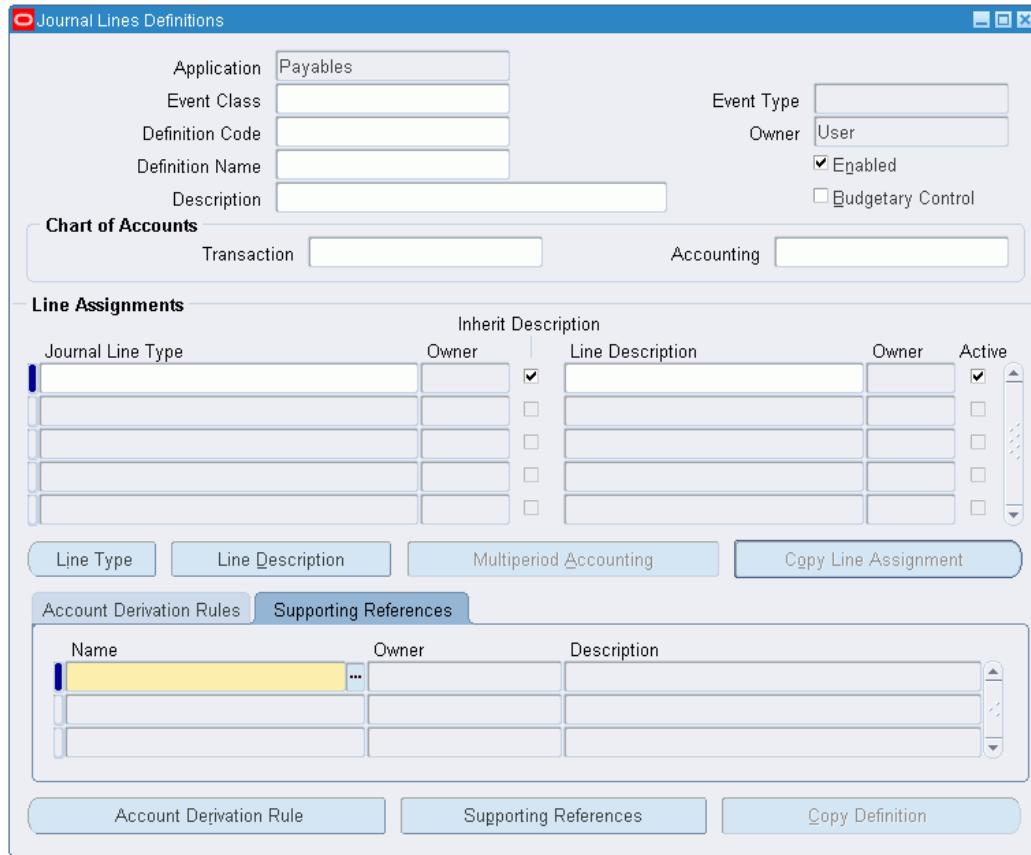
### **Prerequisites**

Define the following:

- Journal line types, page 2-29
- Journal entry descriptions, page 2-48
- Account derivation rules, page 2-57

- Supporting References, page 2-76

## To Define Journal Lines Definitions



This procedure describes selected fields in the Journal Lines Definitions window

1. Navigate to the Journal Lines Definitions window, and in the Find Journal Lines Definitions window, click **New**.

The application name defaults from the application associated with the responsibility.

The Owner field is automatically populated. For components seeded by Oracle, the value is Oracle. For components created on site by users, the value is User.

2. Retain the default for the Enabled check box, which is selected, to make the journal lines definition available for use for this application and the selected chart of accounts.
3. To indicate that the journal lines definition is to be used only if the Create Accounting program is run in budgetary control mode, select the Budgetary Control check box.

4. To assign a journal line type to an event type, in the Journal Line Type field, select a journal line type.

The Owner field is automatically populated based on the journal line type selected.

5. To inherit the journal line description from another journal line, select the Inherit Description check box.

This check box is enabled only if the business flow method of the journal line entry is Prior Entry or Same Entry. If this check box is selected, the Line Description and Owner fields are disabled and cleared of any data.

6. In the Line Description field, enter the journal entry description to be used to populate the subledger journal entry lines.

The list of values includes all journal entry descriptions for the application to which the event class belongs that meets the following condition:

- All sources used by the journal entry description have been assigned to the event class associated with the journal lines definition.

The Owner field is automatically populated based on the line description entered.

7. To use this line assignment to generate subledger journal entries, select the Active check box.

8. In the Line Assignments region, select an accrual journal line type and click **Multiperiod Accounting Assignment** to define the multiperiod journal entry and the multiperiod options.

See: To Define Multiperiod Accounting, page 2-95

9. Select the Account Derivation Rules tab.

10. In the Segment field, select an Accounting Flexfield segment or All Segments.

If entering an accounting chart of accounts for the journal lines definition, the list of values includes all segments in the Accounting Flexfield structure. Otherwise, the list of values includes the following Accounting Flexfield qualifiers:

- All Segments
- Balancing Segment
- Natural Account Segment
- Cost Center Segment
- Management Segment
- Intercompany Segment

If assigning both segment and Accounting Flexfield rules to a journal line type, segment rules take precedence over Accounting Flexfield rules enabling users to overwrite specific segments from the chart of accounts. If defining only qualifier segments in the Account Derivation Rules tab, the rule assignment is not complete. Define an assignment for All Segments.

Alternately, select a segment and build the Accounting Flexfield one segment at a time. A separate account derivation rule must be applied to each segment.

11. If the business flow method is Same Entry for the journal line type, to inherit the value in the Segment field from one side (debit or credit) of the current entry to the other, select the Inherit check box.

If this field is selected, the Rule Name, Owner, Description, and Side fields are blank.

12. In the Rule Name field, select the account derivation rule to populate the complete Accounting Flexfield or a particular segment based on the value for the Segment field.

The list of values includes all account derivation rules associated with the application for which the journal lines definition is created which meet the following conditions:

- All sources used by the account derivation rule are assigned to the event class associated with the journal lines definition.
- If the value All Segments is selected in the Segment field, the list of values includes only Accounting Flexfield rules.

For segment rules, account derivation rules that map to the particular segment or account derivation rules with a value set that maps to the particular segment, the accounting chart of accounts matches the one entered for the journal lines definition and the segment name matches the one entered in the Segment field. If no accounting chart of accounts is specified, the Accounting Flexfield qualifier associated with the segment rule matches the one entered in the previous field.

The Owner and Description fields are automatically populated based on the account derivation rule selected.

The list of values includes the following:

- All account derivation rules associated with the application for which the journal lines definition is created
- Sources that belong to the event class for which the line assignment is created and which meet the conditions described in the table below.

| <b>Transaction Chart<br/>of Accounts</b> | <b>Accounting Chart<br/>of Accounts</b> | <b>Segment</b>                  | <b>Rule Name</b>   |
|--|---|---------------------------------|--|
| Null                                     | Null                                    | All Segments                    | List of values includes all account derivation rules with no transaction and accounting chart of accounts and if the output type of the account derivation rule is Flexfield.  |
| Null                                     | Null                                    | Accounting Flexfield Qualifiers | List of values includes all account derivation rules with no transaction chart of accounts and if the output type of the account derivation rule is the Accounting Flexfield qualifier.  |
| Null                                     | Not Null                                | All Segments                    | List of values includes all account derivation rules with no transaction and null or the same accounting chart of accounts as used by the accounting application definition and if the output type of the account derivation rule is All Segments. |

| <b>Transaction Chart<br/>of Accounts</b> | <b>Accounting Chart<br/>of Accounts</b> | <b>Segment</b> | <b>Rule Name</b>  |
|--|---|----------------|---|
| Null                                     | Not Null                                | Segments       | <p>List of values includes all account derivation rules with no transaction chart of accounts and the following:</p> <ul style="list-style-type: none"> <li>- no accounting chart of accounts if the output type has the same segment</li> <li>- no accounting chart of accounts, and whose output type is value set and that have the same value set as the one used by the Accounting Flexfield segment</li> <li>- same accounting chart of accounts, and whose output type is segment and that have the same segment as the one selected in the Segment field</li> </ul> |
| Not Null                                 | Null                                    | All Segments   | List of values includes all account derivation rules with the same transaction chart of accounts used by the journal lines definition or no transaction chart of accounts and no accounting chart of accounts and whose output type is Flexfield.   |

| <b>Transaction Chart<br/>of Accounts</b> | <b>Accounting Chart<br/>of Accounts</b> | <b>Segment</b>                        | <b>Rule Name</b>  |
|--|---|---------------------------------------|---|
| Not Null                                 | Null                                    | Accounting<br>Flexfield<br>Qualifiers | List of values includes all account derivations rules with the same transaction chart of accounts used by the journal lines definition or no transaction chart of accounts and no accounting chart of accounts and whose output type has the same segment qualifier as the one selected in the Segment field.                   |
| Not Null                                 | Not Null                                | All Segments                          | List of values includes all account derivation rules with the same transaction chart of accounts used by the journal lines definition or no transaction chart of accounts and no accounting chart of accounts or the same accounting chart of accounts used by the journal lines definition and whose output type is Flexfield. |

| Transaction Chart of Accounts | Accounting Chart of Accounts | Segment  | Rule Name   |
|-------------------------------|------------------------------|----------|---|
| Not Null                      | Not Null                     | Segments | <p>List of values includes all account derivation rules with the same transaction chart of accounts used by the journal lines definition or no transaction chart of accounts and the following:</p> <ul style="list-style-type: none"> <li>- no accounting chart of accounts and whose output type has an Accounting Flexfield qualifier that matches the one assigned to the segment</li> <li>- no accounting chart of accounts and whose output type has a value set that matches the value set used by the segment</li> <li>- same accounting chart of accounts and whose output type is the same segment</li> </ul> |

13. Select the Supporting References tab to assign a supporting reference to the journal line type.
14. In the Name field, select the appropriate supporting reference.

The Owner and Description fields are automatically populated based on the supporting reference selected.

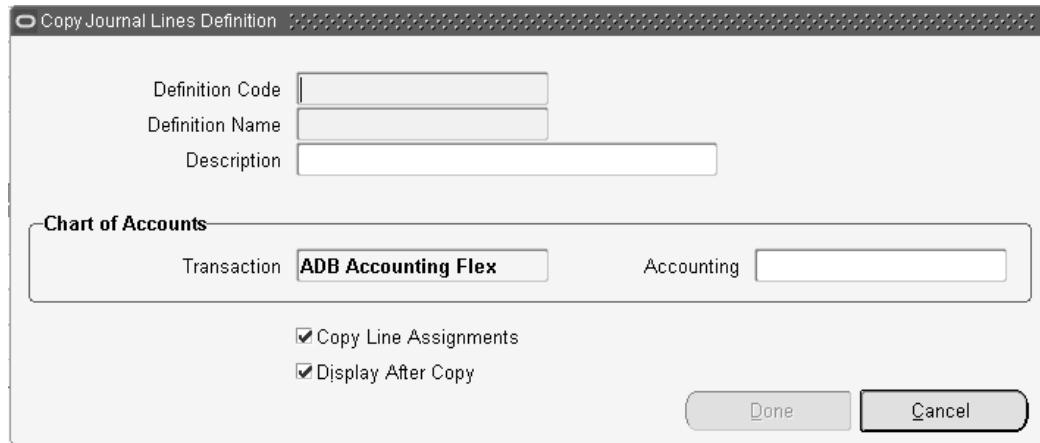
#### Upgrade Account Derivation Rules Tab

Users can enter account derivation rules for Prior-Entry business flow journal line types referring to non-upgraded journal entries. This tab is displayed only for federal installation when the FV: Federal Enabled profile option is enabled.

See: Profile Options, *Oracle U.S. Federal Financials Implementation Guide*

#### To Copy a Journal Lines Definition

See: Copy and Modify Functionality, page 2-6



#### ***Selected Fields in the Copy Journal Lines Definition Window***

| Field                 | Description  |
|-----------------------|--|
| Transaction           | Optionally, if the journal lines definition that is being copied has no transaction chart of accounts assigned to it, select a transaction chart of accounts; otherwise, the transaction chart of accounts defined in the Journal Lines Definitions window defaults to this field. |
| Accounting            | Optionally, if the journal lines definition that is being copied has no accounting chart of accounts assigned to it, select an accounting chart of accounts; otherwise, the accounting chart of accounts defined in the Journal Lines Definitions window defaults to this field.   |
|                       | <b>Note:</b> If there is no accounting chart of accounts, users cannot copy the journal lines definition to a specific chart of accounts if qualifiers used in the account derivations rules do not have a segment in the new accounting chart of accounts.                        |
| Copy Line Assignments | Copies all line assignments to the journal lines definition  |
| Display After Copy    | Automatically displays the new journal lines definition  |

## **Copying Journal Lines Definitions Across Chart of Accounts**

You cannot copy the Journal Line Definitions (JLD) across Chart of Accounts as the line assignments are dependent on the Transaction and Accounting Chart of Accounts. For example, two Journal Line Types (Test Item Expense and Test Item Liability) are created with both the Transaction and Accounting Chart of Accounts (COA) as Operations Accounting Flex. And, two Account Derivation Rules (Test Item Expense Rule and Test Item Liability Rule) are defined with the same Transaction and Accounting chart of account attached to it.

Now define the journal lines definitions with different Transaction and Accounting chart of account, with Services Accounting Flex, then the above two journal lines definitions and account derivation rules are not available in the Line Assignments or the Account Derivation Rules tab. Therefore, when you define the journal lines definitions (Test Accrual) with the Transaction and Accounting chart of account as same as the above journal lines definition and account derivation rules that is Operations Accounting Flex and assign the two journal lines definitions and account derivation rules respectively.

The same rule applies to the Copy functionality. When you copy the journal lines definitions along with the Line Assignments, you cannot change the Transaction and Accounting chart of accounts because the copied journal line definitions and account derivation rules are created with the Transaction and Accounting chart of account of the old Journal Lines Definition. To copy the Journal Lines Definition with different Transaction and Accounting Chart of Account, then copy the Journal Lines Definition without enabling the Copy Line Assignments option. In this case, the application copies only the header information without line assignments and you can enter different Transaction and Accounting Chart of Accounts such as Services Accounting Flex.

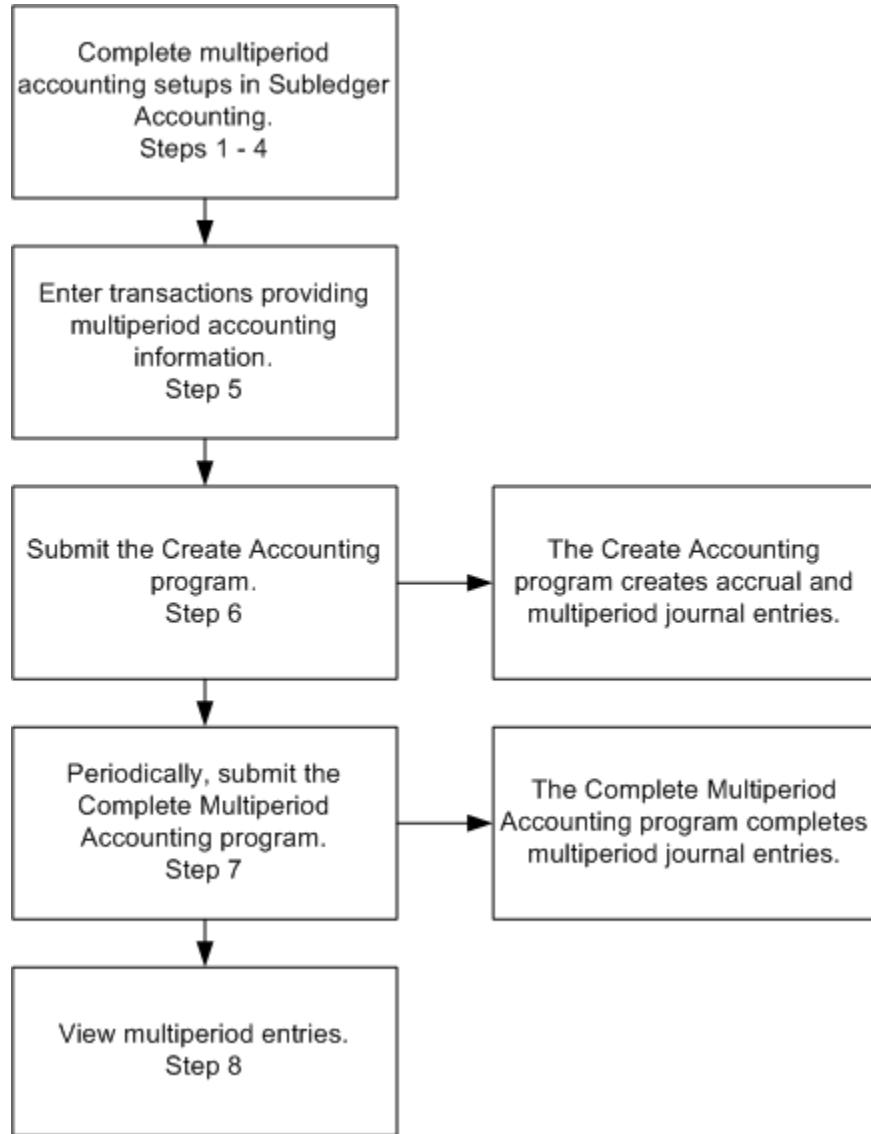
## **Multiperiod Accounting**

Multiperiod accounting enables users to create accounting for a single accounting event for more than one GL period. The functionality is primarily used to recognize revenue or a prepaid expense or revenue across multiple GL periods.

### **Multiperiod Accounting Process**

The figure below is described in Multiperiod Accounting Process Steps, page 2-91.

### **Multiperiod Accounting Process Flow**



### **Multiperiod Accounting Process Steps**

The multiperiod accounting process includes the following steps:

1. In the Journal Line Accounting Attributes Assignments window, assign sources for the following accounting attributes:
  - Multiperiod: Specifies whether a journal line type is eligible for multiperiod accounting
  - Multiperiod Start Date: Date to begin multiperiod accounting

- Multiperiod End Date: Date to end multiperiod accounting

See: To Define Journal Line Types, page 2-31

2. In the Journal Line Types window, define the journal line types to be used for multiperiod accounting by setting the multiperiod option.

See: To Define Journal Line Types, page 2-31

3. In the Journal Lines Definitions window, complete the following:

- Assign the accrual journal line type
- Navigate to the Multiperiod Accounting window to define the multiperiod options

In the Multiperiod Accounting window, users define the multiperiod options.

See: Journal Lines Definitions, page 2-79

4. In the Line Assignments window, define the multiperiod line assignments as follows:

- Update the line description for the accrual journal entry line type, which defaults from the Journal Lines Definitions window
- Assign the recognition journal line types
- Assign account derivation rules

5. Create transactions in subledger applications.

For example, a company may receive a Payables invoice in January for the cost of services for a whole year. However, the company must recognize the expense during the period the expense is incurred. The company enters the originating invoice in Payables to record its receipt.

6. Submit the Create Accounting program to create the originating journal entry as well as the multiperiod entries.

7. Periodically, submit the Complete Multiperiod Accounting program to complete incomplete recognition journal entries as their periods become open.

8. Once the multiperiod journal entries are processed, view the multiperiod journal entries in the inquiry pages for subledger journal entry headers, subledger journal entry lines, and accounting events.

## Multiperiod Accounting Example

A company must prepay rent for the year. \$1,200,000.00 must be paid on December 31, 2005. However, the company needs to recognize the expense over a twelve month period as shown in the table below.

| <b>Total Expense</b> | <b>/</b> | <b>Number of Months</b> | <b>=</b> | <b>Amount Recognized Each Month</b> |
|----------------------|----------|-------------------------|----------|-------------------------------------|
| \$1,200,000.00       | /        | 12                      | =        | \$100,000.00                        |

Assume that the multiperiod accounting duration is from 1-Jan-06 to 31-Dec-06 and that the multiperiod accounting options are as follows:

- Number of journal entries: one per period
- GL Date: first day of the GL period
- Proration method: 360 days

The table below describes the schedule to expense \$100,000.00 each month over the next twelve months.

| <b>Period</b>  | <b>Amount</b> |
|----------------|---------------|
| January 2006   | \$100,000.00  |
| February 2006  | \$100,000.00  |
| March 2006     | \$100,000.00  |
| April 2006     | \$100,000.00  |
| May 2006       | \$100,000.00  |
| June 2006      | \$100,000.00  |
| July 2006      | \$100,000.00  |
| August 2006    | \$100,000.00  |
| September 2006 | \$100,000.00  |

| <b>Period</b> | <b>Amount</b> |
|---------------|---------------|
| October 2006  | \$100,000.00  |
| November 2006 | \$100,000.00  |
| December 2006 | \$100,000.00  |

The following journal entry is created with GL Date 31-DEC-2005 to record the prepaid rent expense.

| <b>Account</b>          | <b>Entered DR</b> | <b>Entered CR</b> | <b>Account DR<br/>(USD)</b> | <b>Account CR<br/>(USD)</b> |
|-------------------------|-------------------|-------------------|-----------------------------|-----------------------------|
| Prepaid Rent<br>Expense | \$1,200,000.00    |                   | \$1,200,000.00              |                             |
| Liability               |                   | \$1,200,000.00    |                             | \$1,200,000.00              |
| Total:                  | \$1,200,000.00    | \$1,200,000.00    | \$1,200,000.00              | \$1,200,000.00              |

The following journal entry with GL date 01-JAN-2006 recognizes the rent expense for January.

| <b>Account</b>          | <b>Entered DR</b> | <b>Entered CR</b> | <b>Account DR<br/>(USD)</b> | <b>Account CR<br/>(USD)</b> |
|-------------------------|-------------------|-------------------|-----------------------------|-----------------------------|
| Rent Expense            | \$100,000.00      |                   | \$100,000.00                |                             |
| Prepaid Rent<br>Expense |                   | \$100,000.00      |                             | \$100,000.00                |
| Total:                  | \$100,000.00      | \$100,000.        | \$100,000.00                | \$100,000.00                |

During the ensuing eleven months, the journal entry shown in the table above will be created to recognize the monthly rent expense as it is incurred.

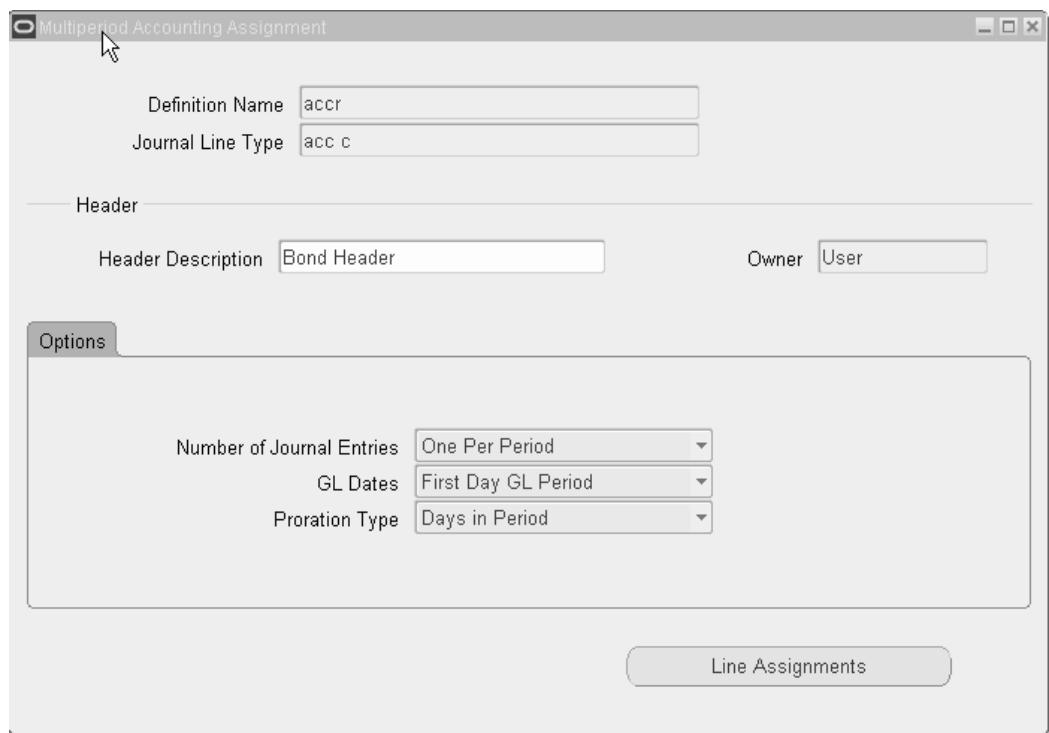
## Defining Multiperiod Accounting

### Prerequisites

Define the following:

- Journal line types, page 2-31
- Journal line accounting attributes, page 2-31
- Journal entry descriptions, page 2-49
- Account derivation rules, page 2-64
- Journal lines definitions, page 2-82

### To Define Multiperiod Accounting



1. In the Line Assignments region of the Journal Lines Definitions window, select the accrual journal line type which will create the originating entry and click **Multiperiod Accounting**.

The Definition Name and Journal Line Type fields default from the Journal Lines Definitions window.

2. Enter data as described in the table below.

| Field                     | Description  |
|---------------------------|--|
| Header Description        | Description for the header of recognition entries  |
| Number of Journal Entries | <p>Number of multiperiod journal entries to be created.<br/>Options are:</p> <ul style="list-style-type: none"><li>• One: One multiperiod journal entry is created to recognize the entire accrual amount.<br/><br/><b>Note:</b> If this option is selected, it is not possible to select a proration type since only one journal entry is created for the entire accrual amount.</li><li>• One per GL Period: One multiperiod journal entry is created per GL period between the dates defined by the multiperiod start and end date accounting attributes.</li></ul> |

---

| Field    | Description  |
|----------|--|
| GL Dates | <p>Specifies when the multiperiod journal entries are created in the GL period</p> <p><b>Note:</b> If a transaction calendar is assigned to the ledger, Subledger Accounting uses this calendar to ensure that the multiperiod accounting journal entries fall on business days.</p> |

Options are:

- First Day GL Period: Multiperiod journal entries have a GL date of the first day of the period. For example, if a company uses monthly periods, the date is October 1.
- Last Day GL Period: Multiperiod journal entries have a GL date of the last day of the period. For example, if a company uses monthly periods, the date is October 31.
- Originating Day: Multiperiod entries have the same GL date as the originating journal entry. For example, if the originating entry has a GL date of October 22, the multiperiod entries have GL dates of the 22nd of the ensuing GL periods, such as November 22 and December 22.

**Note:** If the Originating day option is selected and the originating entry falls on a date that does not occur in every GL period, the closest day within that period is used. For example, if the originating entry has a GL date of October 31, the next multiperiod journal entry would have a GL date of November 30.

---

| Field          | Description  |
|----------------|--|
| Proration Type | <p>Specifies how the recognition amount per period should be calculated. Options are:</p> <ul style="list-style-type: none"> <li>• 360 days: Partial periods calculated by assuming 360 days in a year</li> <li>• Days in Period: Partial periods calculated by dividing the number of partial period days by the actual number of days in the period</li> <li>• First Period: Accrual amount divided by the number of periods and recognized evenly starting with the first period</li> <li>• Total Days in Period: Both the partial periods and the whole periods are based on the actual number of days in the period. The number of days within each period is then divided by the total number of days in the whole multiaccounting period. The result is then multiplied by the total amount due to arrive at an amount for each respective period.</li> </ul> |

**3. Click **Line Assignments**.**

The Definition Name, Owner, accrual Journal Line Type, and Line Description fields default from the Journal Lines Definitions window.

The first accrual journal line creates a mirror image of the original accrual line, reversing the impact of the original journal entry to the same General Ledger account. Users cannot change this journal line type or the account derivation rules for it.

The Inherit check box for the description defaults from the Inherit check box in the Journal Lines Definitions window for the accrual journal line type. If selected, users can deselect the check box and assign a new description.

**4. To assign journal line types to create the recognition journal line, enter data as described in the table below.**

#### ***Selected Fields in the Line Assignments Window***

| <b>Field</b>                          | <b>Description</b>   |
|---------------------------------------|--|
| Journal Line Type                     | Select the corresponding recognition journal line type. The list of values includes only recognition journal line types with the same natural side and event class as the accrual journal line type. |
| Name (Account Derivation Rule region) | The Owner and Description fields default based on the rule selected.   |

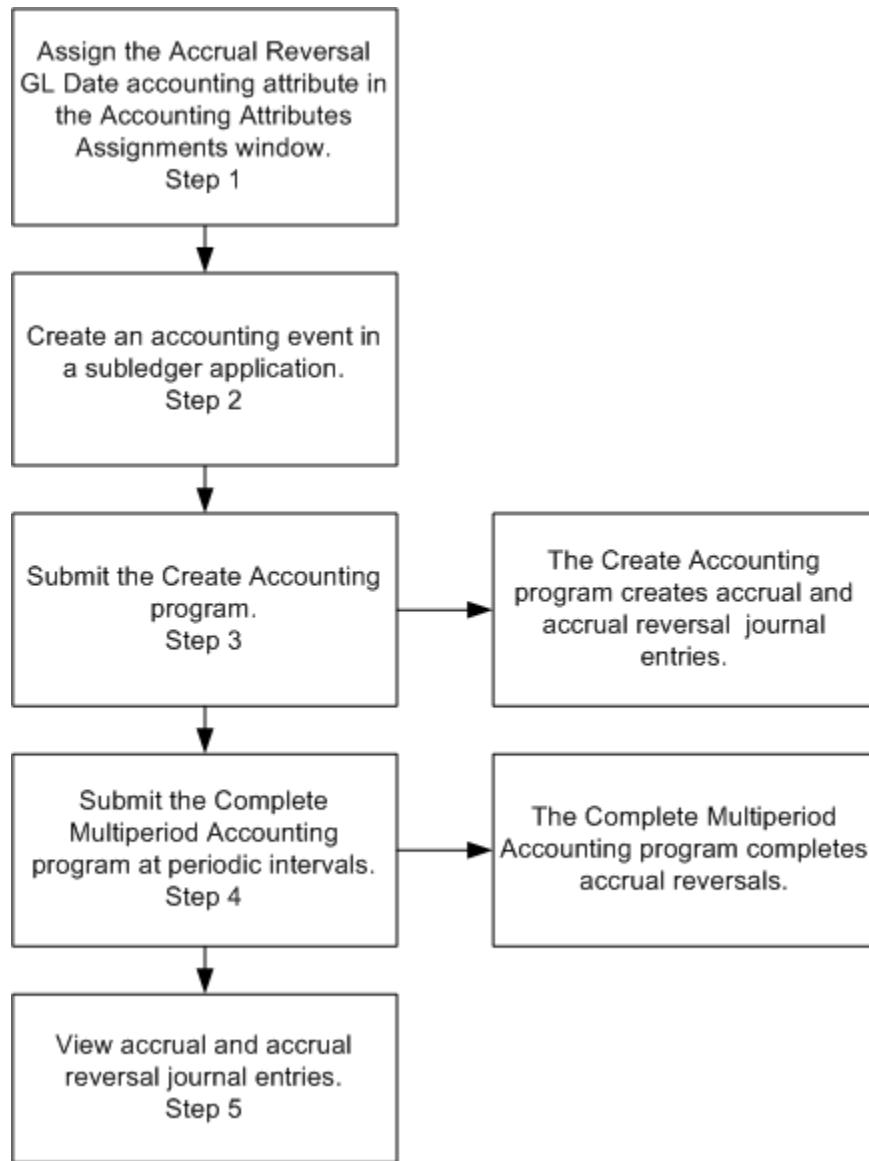
## **Accrual Reversals**

Use accrual reversal to define how and when accrual reversals are automatically performed. You can:

- Indicate that an accounting event is eligible for accrual reversal
- Schedule the reversal of accrual journal entries generated by the Subledger Accounting program

## **Accrual Reversal Process**

The diagram below shows the accrual reversal process and is described in the Accrual Reversal Process Steps, page 2-100.



## Accrual Reversal Process Steps

1. Assign the Accrual Reversal GL Date accounting attribute at the event class level in the Accounting Attribute Assignments window.

Use this attribute to schedule the automatic reversal of a journal entry at the time it is created. Assign any standard date source or one of the following system sources to the accrual reversal GL date accounting attribute:

- Next Day: The GL date of the accrual reversal will be the next day following the GL date of the accrual entry. If there is a transaction calendar assigned to the ledger, this will be the next business day.

- First Day Next GL Period: The GL date of the accrual reversal entry will be the first day of the following period. If there is a transaction calendar assigned to the ledger, this will be the first business day.
- Last Day Next GL Period: The GL date of the accrual reversal entry will be the last day of the following period. If there is a transaction calendar assigned to the ledger, this will be the last business day.

**Note:** Users can override the accrual reversal GL date accounting attribute values on the journal entry header in the Application Accounting Definitions window.

2. Create an accounting event in the subledger application.
3. Submit the Create Accounting program.

The Create Accounting program creates the accrual journal entry as well as accrual reversals and multiperiod journal entries. The program creates the accrual reversal entry to negate the impact of the accrual entry. Depending on the selection for Reversal Method for the ledger accounting options in the Update Accounting Options window, the accrual reversal entry will have the debit and credit signs reversed or it will have negative amounts.

The status of the accrual reversal journal entry and multiperiod journal entry is based upon the status of the GL period of the GL date of the accrual reversal journal entry and the End GL Date specified in the Create Accounting request parameters as described in the table below.

| GL Date  | Open or Future Entry GL Period   | Closed or Permanently Closed GL Period  | Never Opened GL Period  |
|--|--|---|---|
| The GL Date of the accrual reversal is on or before the End GL Date in the mode specified in the Create Accounting request | Creates the journal entry with the status specified in the Create Accounting request | <ul style="list-style-type: none"> <li>• Adjusts the GL date to the next open period</li> <li>• Creates the journal entry with an adjusted GL date in the mode specified in the Create Accounting request if the GL date is prior to the end date of the request or with an incomplete if the date is after the end GL date of the request</li> </ul> | <ul style="list-style-type: none"> <li>• Adjusts the GL date to the next open GL period</li> <li>• Creates the journal entry with an adjusted GL date in the mode specified in the Create Accounting request if the GL date is prior to the end date of the request or as incomplete if the date is after the end GL date in the Create Accounting request</li> </ul> |

| GL Date   | Open or Future Entry GL Period                      | Closed or Permanently Closed GL Period   | Never Opened GL Period                              |
|---|---|--|---|
| GL Date of the accrual reversal is after the END GL Date specified in the Create Accounting request | Creates the journal entry with an incomplete status | <ul style="list-style-type: none"> <li>Adjusts the GL date to the next open GL period</li> <li>Creates the journal entry with an adjusted GL date in the mode specified in the Create Accounting request if the GL date is prior to the end date of the request or with an incomplete status if the date is after the end GL date of the request</li> <li>If no open GL period is found, creates the journal entry with an error status</li> </ul> | Creates the journal entry with an incomplete status |

See: Setting Up Subledger Accounting Options, page 1-2

4. Submit the Complete Multiperiod Accounting program to complete accrual reversals in future GL periods as the periods are opened.

The Complete Multiperiod Accounting program looks at the GL dates of all incomplete accrual reversal and multiperiod journal entries as specified in the Complete Multiperiod Accounting request parameters. It completes those entries with GL dates in open or future entry periods that fall on or before the End GL Date parameter specified in the Complete Multiperiod Accounting request.

Once an accrual reversal or multiperiod journal entry is completed, it is sequenced and appears on reports and inquiries.

See: To Complete Multiperiod Journal Entries and Accounting Reversals, page 2-106

5. View accrual reversals.

See: Subledger Journal Entry Lines Inquiry, page 7-5 and Subledger Accounting

## Accrual Reversal Examples

The following examples describe how accrual reversals are scheduled and entered in journals.

### Example 1

A company receives materials worth \$100 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<br>(USD) | Accounted CR<br>(USD) |
|-------------------|------------|------------|-----------------------|-----------------------|
| Accrual Expense   | 100.00     |            | 100.00                |                       |
| Accrual Liability |            | 100.00     |                       | 100.00                |

The accrual reversal GL date is First Day Next GL period and the following journal entry is created to reverse the accrual.

| Account           | Entered DR | Entered CR | Accounted DR<br>(USD) | Accounted CR<br>(USD) |
|-------------------|------------|------------|-----------------------|-----------------------|
| Accrual Liability | 100.00     |            | 100.00                |                       |
| Accrual Expense   |            | 100.00     |                       | 100.00                |

### Example 2

Futures 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. In this scenario, the investor or ledger must mark the account to market each day and the entry booked from the day before may need to be reversed to reflect the new position.

The following journal entry is created on June 1, 2006.

**GL Date: 01-Jun-2006**

| Account          | Entered DR | Entered CR | Accounted DR<br>(USD) | Accounted CR<br>(USD) |
|------------------|------------|------------|-----------------------|-----------------------|
| Loss             | 100.00     |            | 100.00                |                       |
| Margin Liability |            | 100.00     |                       | 100.00                |

The Accrual Reversal GL Date is set to Next Day and the following journal entry is created to reverse the journal entry from June 1.

**GL Date: 02-Jun-2006**

| Account          | Entered DR | Entered CR | Accounted DR<br>(USD) | Accounted CR<br>(USD) |
|------------------|------------|------------|-----------------------|-----------------------|
| Margin Liability | 100.00     |            | 100.00                |                       |
| Loss             |            | 100.00     |                       | 100.00                |

On June 2, a new journal entry is created to reflect the new position, which will be reversed on June 3.

**GL Date: 02-Jun-2006**

| Account          | Entered DR | Entered CR | Accounted DR<br>(USD) | Accounted CR<br>(USD) |
|------------------|------------|------------|-----------------------|-----------------------|
| Loss             | 105.00     |            | 105.00                |                       |
| Margin Liability |            | 105.00     |                       | 105.00                |

On June 3, the following journal entry is created to reverse the accrual from June 2.

**GL Date: 03-Jun-2006**

| Account          | Entered DR | Entered CR | Accounted DR<br>(USD) | Accounted CR<br>(USD) |
|------------------|------------|------------|-----------------------|-----------------------|
| Margin Liability | 105.00     |            | 105.00                |                       |
| Loss             |            | 105.00     |                       | 105.00                |

## To Complete the Multiperiod Journal Entries and Accrual Reversals

The Complete Multiperiod Accounting program checks the GL dates of all incomplete journal entries that have a GL date that is on or before the end GL date specified in the request parameters and completes these entries as their GL dates fall into open periods.

Enter parameters as described in the table below.

| Parameter                  | Description   |
|----------------------------|---|
| Ledger                     | Limits incomplete journal entries selected for processing to those of the selected ledger; required |
| Process Category           | Restricts events for accounting to the selected category  |
| End GL Date                | End date for processing all incomplete journal entries; required                                    |
| Errors Only                | Limits the completion of journal entries to those processed in error; required                      |
| Report                     | Determines whether to print the report in summary or detail mode                                    |
| Transfer to General Ledger | Determines whether to transfer completed journal entries to General Ledger                          |
| Post in General Ledger     | Determines whether to post completed journal entries transferred to General Ledger                  |

| Parameter                 | Description  |
|---------------------------|--|
| General Ledger Batch Name | Batch name for General Ledger entries posted by the Multiperiod Accounting program |

## Application Accounting Definitions

Use application accounting definitions to assign journal lines definitions, supporting references, and header descriptions to event classes and event types.

Storing the accounting definitions validation status at the event class and event level enables you to generate subledger journal entries for certain event classes or event types even if the accounting definitions for other event classes or event types are invalid.

Each event class and event type assignment consists of a header assignment and one or more journal lines definition assignments. A header assignment includes the following:

- source assignments for the GL date and Accrual Reversal GL date, if enabled for the event class
- a journal entry description (optional)
- one or more supporting references (optional)

You can assign multiple journal lines definitions to an event class or event type.

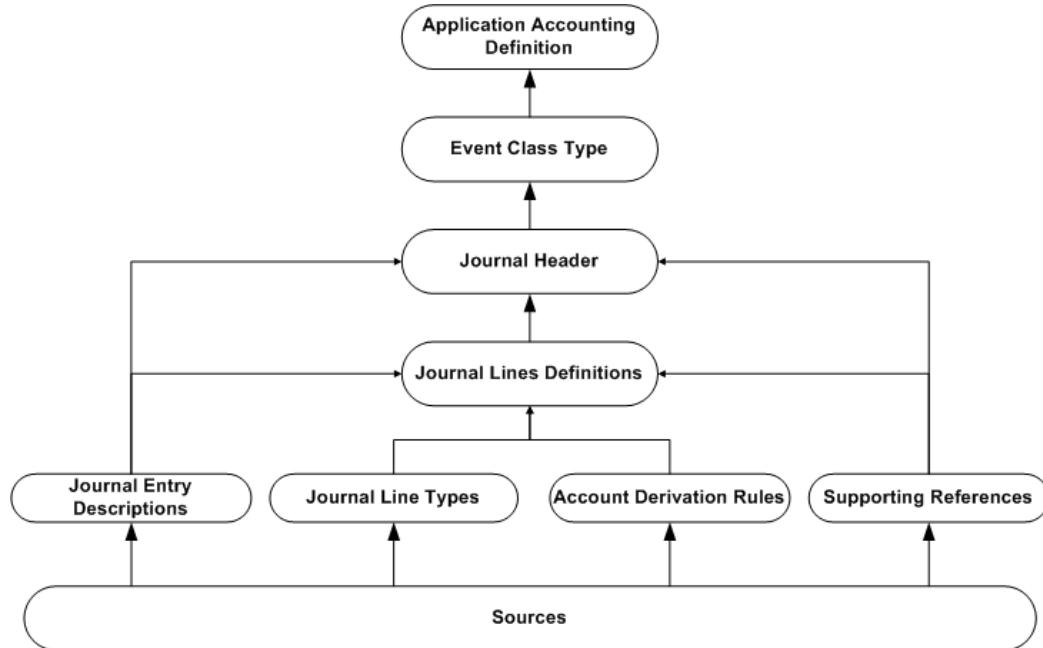
Subledger Accounting generates a single journal entry per accounting event and ledger using the line assignments from all the journal lines definitions assigned to the event class or event type. The following can be assigned to a journal lines definition:

- Journal entry description
- Journal line type
- Account derivation rules
- Supporting references

Sources are used by all of the above components.

The figure below shows the relationship of the components to an application accounting definition as described in the above text.

### ***Application Accounting Definitions***



Application accounting definitions enable you to meet the subledger accounting requirement of multiple accounting representations. While one application accounting definition can generate subledger journal entries that are compliant with one particular set of accounting requirements, another definition can be defined to meet a completely different set of accounting requirements.

For example, use a complete set of US GAAP accounting definitions for Payables as an application accounting definition for the ledger US applications. Use a complete set of French GAAP accounting definitions for Payables can be used for the ledger French Operations. These two sets of definitions have differences based on the setup of the various components that make up their application accounting definitions.

Seeded application accounting definitions are provided for all Oracle subledgers. If specific requirements are not met by startup accounting definitions, users can copy and modify the seeded definitions and their assignments.

The Applications Accounting Definitions (AAD) Loader enables users to import and export application accounting definitions and journal entry setups between the file system and database instances. The AAD Loader also supports concurrent development and version control of the application accounting definitions.

## **Defining an Application Accounting Definition**

Use the Application Accounting Definitions window to assign header descriptions and journal lines definitions to the event classes and event types for a particular application.

Assign a header description to the event class or event type in the Header Assignments

window.

Journal Lines Definitions can be assigned to the application accounting definition in the Application Accounting Definition window.

## Prerequisite

Define the following:

- Journal entry descriptions, page 2-48
- Journal lines definition, page 2-79
- Supporting References, page 2-76

## To Define an Application Accounting Definition

The screenshot shows the 'Application Accounting Definitions (SLA Testing)' window. The 'Application' field is set to 'Receivables'. The 'Owner' field is set to 'Oracle'. The 'Enabled' checkbox is checked. The 'Chart of Accounts' section contains 'Transaction' and 'Accounting' fields. The 'Event Class and Event Type Assignments' section is a grid with columns: Event Class, Event Type, Validation Status, Create Accounting, and Locked. The 'Journal Lines Definition Assignments' section is a grid with columns: Journal Lines Definition Name, Owner, and Description. At the bottom are buttons for 'Journal Line Definition', 'Validate', and 'Copy'.

**Selected Fields and Buttons in the Application Accounting Definitions Window**

---

| Field and Button | Description  |
|------------------|--|
| Application      | Defaults from the application associated with the responsibility   |
| Owner            | Automatically populated. For components seeded by Oracle, the value is Oracle. For components created on site by users, the value is User.   |
| Enabled          | If selected, makes this application accounting definition available  |
| Accounting       | If selected, the following rules apply: <ul style="list-style-type: none"><li>• The application accounting definition can be assigned only to a subledger accounting method with the same chart of accounts.</li><li>• Only those components, such as journal entry descriptions, defined with the same or no charts of accounts are available for assignment in the Header Assignments window.</li></ul> See: Transaction and Accounting Charts of Accounts, page 2-7 |

---

| Field and Button  | Description   |
|-------------------|---|
| Validation Status | <p>Displays the validation status of the event class and event type. Values include:</p> <ul style="list-style-type: none"> <li>• Not Validated: The journal lines definitions assigned to the event class and event type have not been validated.</li> <li>• Valid: All the journal lines definitions assigned to the event class and event type have been successfully validated, and the underlying database package has been successfully created.</li> <li>• Invalid: The validation of the journal lines definitions assigned to the event class and event type failed for at least one of them, or the underlying database package could not be created because of a technical problem.</li> </ul>   |
| Create Accounting | <p>An application accounting definition can only be used to create subledger journal entries for a valid event class and event type. The Subledger Accounting program creates errors in the following cases:</p> <ul style="list-style-type: none"> <li>• For the event being processed, the associated event class and event type has an invalid status.</li> <li>• For the event being processed, the associated event class or event type is not included in the application accounting definition.</li> </ul> <p>Select to create subledger journal entries for the accounting events associated with the event class and event type.</p> <p>If this check box is deselected, no journal lines definitions can be assigned to the event class and event type. If any journal lines definitions are assigned to the event class and event type, this check box cannot be deselected.</p> |

| Field and Button              | Description  |
|-------------------------------|--|
| Locked                        | Controls whether the application accounting definition can be modified   |
| Header Assignments            | <p>Opens the Header Assignments window to assign a journal entry description and supporting references to the event class and event type of the application accounting definition and to view the accounting attributes associated with the journal entry header.</p>                |
|                               | <p><b>Note:</b> Supporting references maintaining balances or with line level sources assigned to the details may not be used on journal entry headers.</p>  |
| Journal Lines Definition Name | <p>Assign ALL - event type journal lines definition to an event class and event type with ALL association in the application accounting definition and journal lines definitions window. List of values includes all journal lines definitions that meet the following criteria:</p> |
|                               | <ul style="list-style-type: none"> <li>The event class and event type for the journal lines definition matches the ones being assigned to the application accounting definition.</li> </ul>  |
|                               | <p>Users can assign one or more journal lines definitions to the event class or event type. Subledger Accounting generates a journal entry per event and ledger using the line assignments from all journal lines definitions assigned to the event class and event type.</p>        |
| Validates                     | Validates the application accounting definition  |

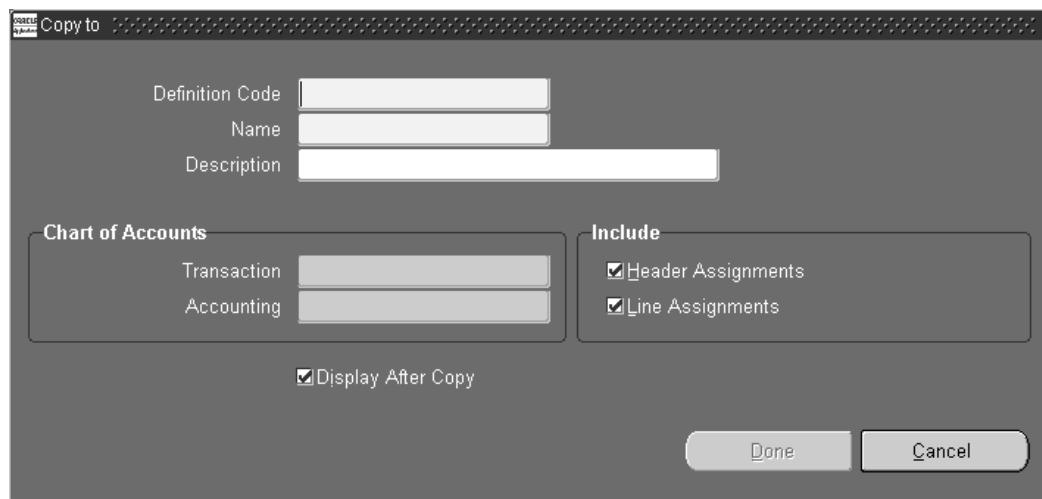
## Copying and Modifying Application Accounting Definitions

See: Copy and Modify Functionality, page 2-6

When copying an application accounting definition, copy the header and line

assignments associated with the original definition, or copy a definition and create the assignments manually.

### To Copy, Modify, and Validate Application Accounting Definitions



#### ***Selected Fields in the Copy to Window***

| Field                    | Description  |
|--------------------------|--|
| Chart of Accounts region | <p>The transaction and accounting charts of accounts default from the original application accounting definition. Note that if values have defaulted for these fields, they cannot be updated. However, if the original application accounting definition does not have a transaction or accounting chart of accounts, then users can enter them in this window.</p> <p>If the charts of accounts are entered, then the following rules apply:</p> <ul style="list-style-type: none"><li>• The new application accounting definition can only be assigned to a subledger accounting method with the same or no charts of accounts.</li><li>• Only those components defined with the same or no charts of accounts, such as account derivation rules and journal line types, are available for assignment in the new application accounting definition.</li></ul> <p>See: Transaction and Accounting Charts of Accounts, page 2-7</p> |
| Include region           | Select the appropriate check boxes to copy the assignments associated with the original application accounting definition to the new one.  |

After modifying a component of an application accounting definition, its validated status changes to Not Validated. Validate the modified definition in the Application Accounting Definitions window. Once validated, use it to generate subledger journal entries.

## **Import Application Accounting Definitions**

Application accounting definitions are imported using the Import Application Accounting Definitions concurrent program. This program imports the application accounting definitions from a data file to the Accounting Methods Builder (AMB) context specified in the SLA: Accounting Methods Builder Context profile option and produces a report of the results.

See: SLA: Accounting Methods Builder Context, page B-3

When running the Import Application Accounting Definitions concurrent program, users specify whether to run the merge analysis, merge, or overwrite processes.

The main purpose of the Import Application Accounting Definitions program is to validate AAD. Run this program after applying any patch changing the rules, as it leaves the AADs invalid.

### **Import Application Accounting Definitions Program Parameters Description**

The table below describes the parameters for the Import Application Accounting Definitions program.

#### ***Import Application Accounting Definitions Program Parameters***

| <b>Parameter</b>                           |  |
|--|--|
| Accounting Methods Builder Context         | AMB context for the application accounting definitions to be imported; defaulted from the SLA: Accounting Methods Builder Context profile option; required                                     |
| Source Data File                           | Full path name, including the .ldt file name, of the data file containing the application accounting definitions to be imported; required<br><br>Path name example:<br>/home/doe/out/APAAD.ldt |
|  | <b>Note:</b> At least one of the three source path names must be entered.  |
| Source File Pathname for Base Application  | Valid source file path name  |
| Source File Pathname for Budgetary Control | Valid source file path name  |
| Merge Analysis Only                        | Determines whether merge analysis is performed to the imported application accounting definitions. Default is Yes.   |
| Batch Name                                 | User-entered merge analysis report name  |

---

| Parameter       |   |
|-----------------|---|
| Import Option   | Enabled and required only if the Merge Analysis Only parameter is set to No; indicates whether the application accounting definitions from the data file should be merged or overwritten to the database. Default is Merge.   |
| Validate        | Enabled and required only if the Merge Analysis Only parameter is set to No; determines whether the Validate Application Accounting Definitions concurrent program should be submitted to validate all the imported application accounting definitions. Default is Yes.   |
| Force Overwrite | Enabled and required only if the Merge Analysis Only parameter is set to No and the Import Option is Overwrite; indicates whether Subledger Accounting should allow an application accounting definition to be overwritten from the data file to the database if the version in the data file is lower than the version in the database. Default is No. |

---

### Import Application Accounting Definitions Report

When the Import Application Accounting Definitions program is completed successfully, a report highlighting all the imported application accounting definitions and their versions as well as all the errors that occurred during the import process is produced.

## Export Application Accounting Definitions

The Export Application Accounting Definitions program exports all application accounting definitions of an application from a database to the file system and produces a report of the results. All application accounting definitions and journal entry setups for an application are exported to the same data file. When the application accounting definitions are exported, a new version is stamped on the application accounting definitions, mapping sets, and account derivation rules referenced by exported application accounting definitions.

## Export Application Accounting Definitions Program Parameters

The table below describes the parameters for the Export Application Accounting Definitions program.

### ***Export Application Accounting Definitions Program Parameters***

| Parameter                                       | Description  |
|---|--|
| Account Method Builder Context                  | AMB context for the application accounting definitions to be imported; defaulted from the SLA Accounting Methods Builder Context profile option; required  |
| Destination File Path                           | Full path name, including the .ldt file name, of the file system where the application accounting definitions are to be exported; required<br><br>Path name example:<br>/home/doe/out/APAAD.ldt<br><br><b>Note:</b> At least one of the three destination file path names must be entered.   |
| Destination File Pathname for Base Application  | Valid file path name for the base application  |
| Destination File Pathname for Budgetary Control | Valid path name for the federal budgetary control  |
| Versioning Mode                                 | Indicates whether the exported application accounting definitions are a result of leapfrogging. Select Standard if the AAD to be exported is based on the latest version; select Leapfrog if the AAD to be exported is not based on the latest version; or select Supersede if the AAD to be exported is not based on the latest version and is not a leapfrog. Default is Standard. |
| User Version                                    | User-assigned version; optional  |
| Export Comment                                  | User-entered export comments; optional   |

## Export Application Accounting Definitions Report

When completed successfully, the Export Application Accounting Definitions program produces a report highlighting all the exported application accounting definitions as well as all the errors that occurred during the export process.

## Subledger Accounting Methods

Application accounting definitions that comply with a common set of accounting requirements can be grouped into a subledger accounting method. The grouping allows a set of application accounting definitions to be assigned collectively to a ledger. This reduces setup time and helps ensure a consistent method of accounting for all subledgers feeding into a particular ledger.

For example, a subledger accounting method entitled French GAAP can be defined to group application accounting definitions that are accounted for using French GAAP criteria. As another example, a Cash Basis Accounting Method can be defined to group application accounting definitions that are used to account for transactions on a cash basis.

By assigning different subledger accounting methods to ledgers, the AMB enables users to create multiple accounting representations of transactions.

See: Multiple Representations, page 2-8

## To Define a Subledger Accounting Method

Subledger Accounting Methods

Method Code:

Method Name:

Description:

Owner: **Oracle**  Enabled

**Chart of Accounts**

Transaction:

Accounting:

**Application Accounting Definition Assignments**

| Application | Name | Context | Owner | Start Date | End Date |
|-------------|------|---------|-------|------------|----------|
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |
|             |      |         |       |            |          |

Description:

**Application Accounting Definition** **Copy Line Assignment**

**Note:** Oracle recommends that users do not modify a seeded method or any other seeded component as it could get overwritten in an upgrade. Instead, copy a seeded component and then modify it appropriately. The modified component has an Owner type of User.

### Selected Fields in the Subledger Accounting Methods Window

| Field | Description  |
|-------|--|
| Owner | Automatically populated. For components seeded by Oracle, the value is Owner. For components created on site by users, the value is User.<br><br>When selecting component names from a list of values in AMB windows, users are presented with the name as well as the Owner of the component. This enables users to distinguish between seeded and user-defined components. |

| Field                    | Description  |
|--------------------------|--|
| Enabled                  | If selected, makes this subledger accounting method available for use  |
| Chart of Accounts region | <p><b>Note:</b> The transaction and accounting chart of accounts are optional. If they are entered, then the following rules apply:</p> <ul style="list-style-type: none"> <li>Only an application accounting definition with the same or no charts of accounts is available for assignment.</li> <li>The accounting chart of accounts must match the chart of accounts of the ledger that this subledger accounting method is assigned to.</li> </ul> <p>See: Transaction and Accounting Charts of Accounts, page 2-7</p> |
| Application              | Application that owns the application accounting definition for this subledger accounting method   |

## Copy and Modify a Subledger Accounting Method

See: Copy and Modify Functionality, page 2-6

When a subledger accounting method is copied, the new method is assigned all the application accounting definitions and associated header and line components of the original subledger accounting method. For example, if the subledger accounting method US GAAP is copied to a new subledger accounting method called US Management, then all the components of the AMB associated with US GAAP are now also associated with US Management.

To customize any of the components associated with the new subledger accounting method, copy the relevant component or create a new one and associate it with the new subledger accounting method.

## To Copy a Subledger Accounting Method



The Transaction and Accounting charts of accounts defaults from the original subledger accounting method. Note that if a value has defaulted for these fields, they cannot be updated. However, if the original subledger accounting method does not have a transaction or accounting chart of accounts, enter them in this window.

These entries serve as the charts of accounts for the subledger accounting method created. If entered, only components defined with the same or no charts of accounts are available for assignment to the new subledger accounting method. The accounting chart of accounts, if entered, must match the chart of accounts of the ledger that this new subledger accounting method is to be assigned to.

See: Transaction and Accounting Charts of Accounts, page 2-7

## Accounting Definitions Inquiry

Accounting definitions inquiry enables users to:

- Identify accounting definitions in which a certain component is included
- Query all application accounting definition assignments by application accounting definition, event class, or event type

Use this inquiry to plan and assess the impact of modifications to the application accounting definitions. For example, you may want to add the customer name to the journal entry description for cash receipts, but they are unsure whether the same journal entry description is being used for miscellaneous receipts which do not have an associated customer. In another scenario, you can change an account derivation rule to take the cost center from a mapping set based upon Oracle Projects expenditure types but are unsure whether the same account derivation rule is used for other cases in which the mapping set might not apply.

## Accounting Definitions Inquiry Windows

### Find Accounting Definitions Window

In the Find Accounting Definitions window, navigate to the following windows:

- Accounting Definition Headers window, described below, when querying accounting header descriptions
- Accounting Definitions Lines window, described below, when querying the following line components:
  - Accounting line type
  - Account derivation rule
  - Accounting line description

### Accounting Definition Headers Window

In this window, navigate to the following windows:

- Accounting Definition Lines window, described below, to examine the line assignments associated with each header
- Application Accounting Definition window, page 2-109  
If the application accounting definition is unlocked, navigate to the corresponding line assignment in the Assignments window.
- Header Assignments window, page 2-109

### Accounting Definition Lines Window

In this window, view the following:

- Assignments
- Journal line types
- Account derivation rules
- Application accounting definitions

From this window, navigate to the following windows:

- Application Accounting Definition window, page 2-109  
If the application accounting definition is unlocked, users can navigate to the

corresponding line assignment in the Assignments window.

- Account Derivation Rules window, page 2-64
- Journal Lines Definitions window, page 2-79 to view line assignments



---

## Create Accounting and Transfer Journal Entries to GL Programs

### Overview of Create Accounting and Transfer Journal Entries to GL Programs

The Create Accounting program processes eligible accounting events to create subledger journal entries. To create the subledger journal entries, the Create Accounting program applies application accounting definitions that are created in the Accounting Methods Builder (AMB).

The Create Accounting program:

- Validates and creates subledger journal entries
- Transfers the final journal entries in the current batch run to General Ledger and starts the General Ledger posting process
- Generates the Subledger Accounting Program Report, which documents the results of the Create Accounting program

The Transfer Journal Entries to GL program enables you to transfer any eligible journal entries to General Ledger, including those from previous batch runs that have not yet been transferred to General Ledger.

If you already created accounting in Final Mode but have not transferred them over to GL, and you run the Create Accounting program with Transfer to GL option, then the application does not pick up events that are already created in the final mode. You run the Transfer Journal Entries to General Ledger program to do that.

### Create Accounting Program

The Create Accounting program creates and transfers journal entries. In general, the parameters described in the table below determine which accounting events are

processed.

#### ***Create Accounting Program Parameters***

| <b>Parameter</b> | <b>Description</b>   |
|------------------|--|
| Ledger           | <p>Required; limits accounting events selected for processing to those of a particular ledger</p> <p>Subledger Accounting lists ledgers that have events that the responsibility can access due to transaction-based security. If the profile option SLA: Enable Data Access Security in Subledgers is set to Yes, only those ledgers that are included in General Ledger Access Sets assigned to the responsibility appear in the list of values.</p> <p>If the application is a valuation method application as defined in the AMB, then both primary and secondary ledgers within the access sets are displayed. If the application is not a valuation method application, then only primary ledgers are displayed.</p> |
| Process Category | <p>Optional; restricts the events selected for accounting to a particular process category</p> <p>The Create Accounting program selects events with event types falling into the event classes assigned to the process category.</p> <p>Only process categories established by development teams as part of the seed data are included in the list of values.</p>  |
| End Date         | <p>Required; end date for the Create Accounting program; processes only those events with event dates on or before the end date</p>  |

---

| Parameter                  | Description  |
|----------------------------|--|
| Mode                       | <p>Required; determines whether the subledger journal entries are created in Draft or Final mode</p>   |
|                            | <p>If Draft mode is selected, then the Transfer to GL, Post in General Ledger, and General Ledger Batch Name fields are disabled. Draft entries cannot be transferred to General Ledger.</p> |
|                            | <p>See: Subledger Accounting Setup Options Description, page 1-3</p>   |
| Errors Only                | <p>Required; limits the creation of accounting to those events for which accounting has previously failed</p>  |
|                            | <p>If Yes is selected, the Create Accounting program processes all events with a status of Error.</p>  |
|                            | <p>See: Subledger Accounting Setup Options Description, page 1-3</p>   |
| Report                     | <p>Required; determines whether to generate a report showing the results of the Subledger Accounting program in summary or detail format</p>   |
|                            | <p>See: Subledger Accounting Setup Options Description, page 1-3</p>   |
| Transfer to General Ledger | <p>Required if Mode is set to Final; determines whether to transfer the subledger journal entries to General Ledger</p>  |
|                            | <p>Journal import is not launched if the profile option SLA: Disable Journal Import is set to Yes.</p>   |
|                            | <p>See: Subledger Accounting Setup Options Description, page 1-3</p>   |

---

| Parameter                            | Description   |
|--------------------------------------|---|
| Post in General Ledger               | Required if Mode is set to Final or Create Accounting is set to No; determines whether to post subledger journal entries in General Ledger  |
|                                      | If there is a primary ledger with secondary ledgers attached to it and the application is a nonvaluation method application, the journal entries of the secondary ledgers are automatically posted. This is based on the application accounting definition that the user sets up. |
|                                      | If the application is a valuation method application, then there is a different event for that secondary ledger and the accounting is not created in the same run for both ledgers.   |
| General Ledger Batch Name            | See: Subledger Accounting Setup Options Description, page 1-3   |
| Include User Transaction Identifiers | Optional; user-entered batch name that appears on the transferred General Ledger subledger journal entries. Transfer to GL option must be set to Yes.   |
|                                      | Required; controls whether the report displays user identifiers' names and values.  |

## Transfer Journal Entries to GL Program

The Transfer Journal Entries to GL program consists of a subset of parameters used in the Create Accounting program as listed below:

- Ledger
- Process Category
- End Date
- Post in General Ledger
- General Ledger Batch Name

Please refer to the descriptions in the Create Accounting Programs Parameters table, page 3-1 .

## Oracle Subledger Accounting Program Report

The Subledger Accounting Program Report is generated by the Create Accounting program. It lists the following:

- Successful events and the subledger journal entries created for those events
- Errors for failed events

Users can run the report in summary or detail mode which are described as follows:

- Summary mode provides a summary of events processed and detailed information about their errors.
- Detail mode provides details of subledger journal entries generated from the processing of completed events and a detailed error report.

## Transfer Journal Entries to GL Report

The Transfer to Journal Entries to GL Report is generated by the Transfer Journal Entries to GL program and lists the following:

- Transfer to GL Summary
- General Errors



---

## Conversion of Historical Data for Reporting Currencies and Secondary Ledgers

### Conversion of Historical Data for Reporting Currencies and Secondary Ledgers

When defining a new subledger reporting currency or secondary ledger for an active primary ledger, you can use the SLA Secondary/ALC Ledger Historic Upgrade program to initialize journals.

You can determine the period from which to create ledger balances by selecting the first period for conversion. The journals are created from the selected period onwards. The first period of historic conversion should be the earliest period for which you have open transactions.

When a subledger level secondary or reporting ledger is added to an existing primary ledger, the accounting in the secondary or reporting ledger fails in the following scenarios:

- When a final accounted transaction is reversed: A Payables invoice is created and final accounted in the primary ledger. Then, a new subledger level secondary or reporting ledger is added to this primary ledger. The same invoice is cancelled. When the Create Accounting program is run, accounting for the invoice cancellation fails for the secondary or reporting ledger, as the accounting program does not find the corresponding Invoice accounting entry to generate the transaction reversal entry (Invoice cancellation accounting) for the secondary or reporting ledger.
- When a Payment accounting entry needs to be created using business flows: A Payables invoice is created and final accounted in the primary ledger. Then, a new subledger level secondary or reporting ledger is added to this primary ledger. A payment is generated against the same invoice. When the Create Accounting program is run, accounting for this payment fails for the secondary or reporting

ledger, as the accounting program cannot find the corresponding upstream entry (Invoice accounting) to generate the downstream entry (Payment accounting) for the secondary or reporting ledger.

A new subledger level secondary/reporting ledger should not be added to an existing primary ledger that already has final accounted subledger journal entries.

## Running the SLA Secondary/ALC Ledger Historic Upgrade Program

Use this procedure to run the SLA Secondary/ALC Ledger Historic Upgrade program.

### Prerequisites:

Create initial balances for new ledgers:

**Note:** The SLA Secondary/ALC Ledger Historic Upgrade program creates journals in Oracle Subledger Accounting. These journals are created with a posting status of Posted and are never actually posted to Oracle General Ledger. To maintain historic balances in Oracle General Ledger, you must initialize balances for the new reporting currency ledger or secondary ledger.

To create initial balances for a:

- Reporting Currency Ledger: Run the Reporting Currency: Create Opening Balance Journals in Reporting Currency program. For more information on running this program, see the Implementation Considerations, *Oracle General Ledger User's Guide* topic within the *Oracle General Ledger User's Guide*.
- Secondary Ledger: Use the Consolidation Workbench to copy initial balances. For more information on using the Consolidation Workbench, refer to the *Oracle Financials Implementation Guide*.

### Restrictions:

- The program cannot be used for a secondary ledger whose subledger accounting method is different from that of the primary ledger.
- The program should be run immediately after creation of the new secondary/reporting currency ledger so that no new transactions are made between the creation of the new secondary/reporting ledger and running of the historic upgrade program.
- All fully accounted transactions of the primary ledger must be transferred to Oracle General Ledger before running the program.

### To run the SLA Secondary/ALC Ledger Historic Upgrade program:

1. Navigate to the **Submit Request** form.
2. Select **SLA Secondary/ALC Ledger Historic Upgrade** from the LOV in the Name field.
3. Enter the parameters as follows:
  1. **Primary Ledger:** Select the primary ledger.
  2. **Reporting/Secondary Ledger:** Select the reporting/secondary ledger.
  3. **Upgrade Start Period:** Select the first period from which you wish to run the historic upgrade.
  4. **Secondary Conversion Option:** Select the secondary conversion option to be used when running a historic upgrade for a secondary ledger. The options are Derive from Original Transaction Rate and Use Initialization Rate.
  5. **Secondary Currency Conversion Type:** Select the currency conversion type or rate type to be used when running a historic upgrade for a secondary ledger.
  6. **Secondary Currency Conversion Date:** Select the conversion date to be used when running a historic upgrade for a secondary ledger.
  7. **Mode:**
    - **Final:** Select this value after you run the program in the Validation mode. This mode creates the journal entries within Oracle Subledger Accounting.
    - **Validation:** Select this value to run the program in Validation mode.
 

**Important:** It is recommended to first run the program in Validation mode so that the program runs the validations that must be checked before performing the historic upgrade. If validations fail, you can view them within the log file and take action. After the program runs successfully in the Validation mode, you can run it in Final mode.
    - **Recovery:** Select this value to purge results after a concurrent request is terminated while in Final mode.
 

**Important:** If you terminate the program while it is running in Final mode, then you must first run the program in Recovery mode before running the program again in Final

mode. Recovery mode purges any upgrade data that may have been created during the terminated concurrent request.

**Note:** In the case of a reporting currency ledger, the conversion option, rate type, and rate date are picked up from the ledger relationships defined in the Accounting Setup Manager.

## Running the Upgrade Historical Subledger Transaction Accounting Program on Demand

Oracle Subledger Accounting provides a concurrent program, Upgrade Historical Subledger Transaction Accounting, that allows the historical accounting data conversion to be done independently for each subledger.

You must run the concurrent program Upgrade Historical Subledger Transaction Accounting instead of using the On Demand Upgrade Patch.

To run the Upgrade Historical Subledger Transaction Accounting, enter the following parameters:

1. Select or enter the Application name. The list shows the application attached to the responsibility from where the concurrent program is submitted.
2. Enter the name of the Ledger. The list shows all Primary ledgers.
3. Enter the Start Period Name. The list shows all periods not yet upgraded and associated with the ledger selected in the above parameter.
4. Enter the Number of Workers. The value should be greater than 1. Otherwise, the application inputs the default value of 1.
5. Enter the Batch size. The default value is 1000.

---

## Diagnostic Framework

### Diagnostic Framework Overview

The diagnostic framework provides features to review the journal entries created by the Subledger Accounting program. It provides the input data used for each transaction, such as the following:

- Transaction objects
- Ledger information
- System sources
- Source values

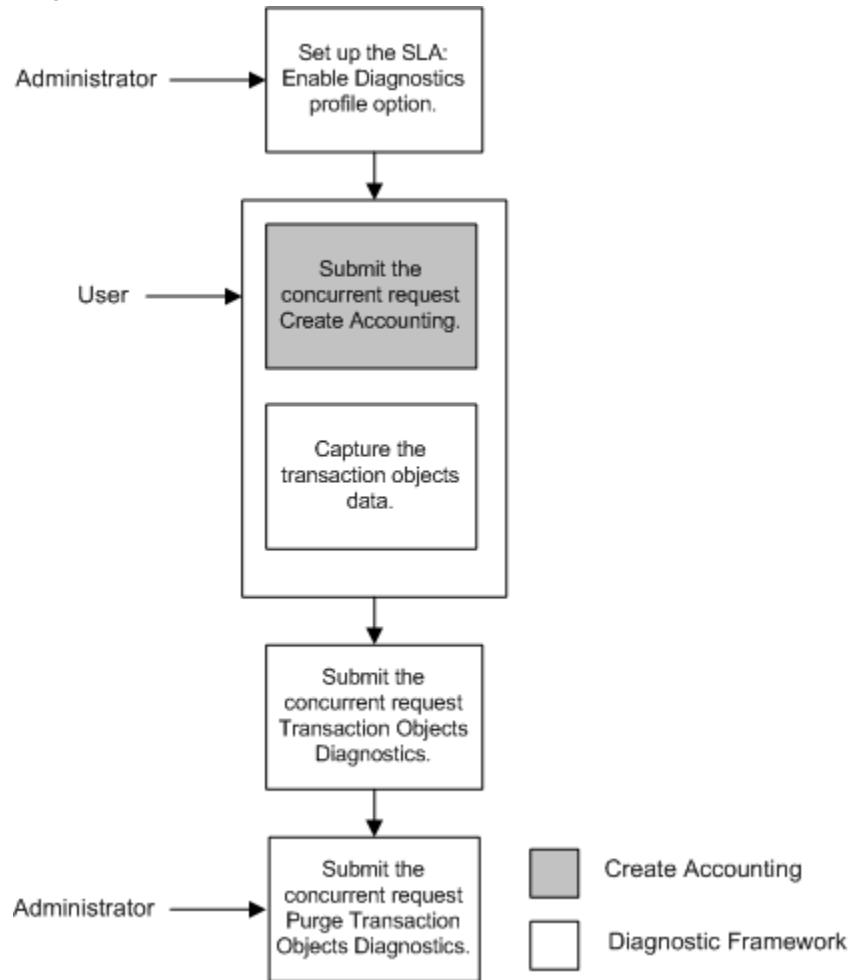
The following examples describe typical uses of the diagnostic framework features.

- In the implementation phase, users can launch the Transaction Objects Diagnostic report to review the source values available in the transaction objects.
- On a daily basis, users can use the Transaction Objects Diagnostic report to analyze issues related to the generation of subledger journal entries.

### Diagnostic Framework Business Process Flow Diagram

The figure below illustrates the business process for the generation and the report of the diagnostic framework output data and is described in the succeeding text.

### Diagnostic Framework Business Process Flow



The following steps describe the diagnostic framework business process flow:

1. The administrator sets the SLA: Enable Diagnostics profile option to Yes for the user or responsibility.  
See: SLA: Enable Diagnostics, page B-5
2. Users submit the Create Accounting program which automatically executes the diagnostic framework to populate the diagnostic tables.
3. The diagnostic framework gathers source values and other information from the transaction objects.
4. Users submit the Transaction Objects Diagnostics concurrent request to view the diagnostic report.
5. The administrator submits the Purge Transaction Objects Diagnostics concurrent

request to purge the results of the diagnostic framework.

## Diagnostic Framework Features

The diagnostic framework features are as follows:

- SLA: Enable Diagnostics Profile Option, page 5-3
- Diagnostic Framework Execution, page 5-3
- Transaction Objects Diagnostics Report, page 5-3
- Purge Transaction Objects Diagnostics, page 5-5

### SLA: Enable Diagnostics Profile Option

The SLA: Enable Diagnostics profile option controls whether diagnostic information is gathered by the Create Accounting program.

See: SLA: Enable Diagnostics, page B-5

### Diagnostic Framework Execution

If the SLA: Enable Diagnostics profile option is set to Yes, the diagnostic framework is executed simultaneously with the Create Accounting program. The diagnostic framework data is stored in the diagnostic tables.

### Transaction Objects Diagnostics Report

To view the diagnostic framework report, users submit the Transaction Objects Diagnostics program with the parameters described in the table below.

### ***Transaction Objects Diagnostics Report Parameters***

| <b>Field</b>             | <b>Description</b>  |
|--------------------------|---|
| Ledger                   | <p>Optional; limits the diagnostic data selected to those of a particular ledger</p> <p>If the SLA: Enable Data Access Security in Subledgers profile option is set to Yes, only those ledgers that are included in the General Ledger Access Sets assigned to the responsibility appear in the list of values. If this profile option is set to No, all ledgers are included.</p> <p>List of values includes primary and secondary ledgers with subledger accounting methods assigned to them.</p>                       |
| Event Class Name         | <p>Optional; limits the diagnostic data to the specified event class within the application</p> <p>List of values includes the event classes for the corresponding application.</p>   |
| Event Type Name          | <p>Optional; limits the diagnostic data to the specified event type within the event class. To enable this event, users enter a transaction number.</p> <p>List of values includes the event types matched with the current application, entity, and event class.</p> <p>If no event class is selected, users cannot enter a value. If an event class is entered, the value from the list of values is dependent on the value of the event class. If the event type field is blank, all the event types are selected.</p> |
| Transaction Number       | <p>Optional; limits the diagnostic data retrieved to the specified transaction. To enable this field, users must enter a transaction number.</p>  |
| Event Number             | <p>Optional; limits the diagnostic data retrieved to the specified event</p>  |
| From Extract Line Number | <p>Optional; displays the data for the transaction object lines that are equal to or higher than the number specified</p> <p>To enable this field, users must enter a transaction number and an event number.</p> <p>List of values includes all transaction object line numbers for the given event.</p>   |

| Field                         | Description  |
|-------------------------------|--|
| To Extract Line Number        | <p>Optional; displays the data for the transaction object lines that are equal to or lower than the number specified</p> <p>To enable this field, users must enter a transaction number and an event number.</p> <p>List of values includes all transaction object line numbers for the given event.</p> |
| Accounting Program Request ID | <p>Optional; restricts the diagnostic data to the specified request ID</p> <p>List of values retrieves the request IDs stored in the diagnostic framework tables.</p>  |
| Errors Only                   | <p>Required</p> <p>If Yes is selected, displays the diagnostic framework data only for the events that meet the selection criteria and whose status is Error. Default is No.</p>   |
| Display Source Name           | <p>Required</p> <p>If Yes is selected, the source name is displayed. If No is selected, the source code is displayed. Default is No.</p>   |
| Display Accounting Attributes | <p>Required</p> <p>If Yes is selected, the accounting attributes are displayed. Default is No.</p>   |

## Purge Transaction Objects Diagnostics

Diagnostic framework data can be purged:

- When a transaction previously accounted in draft mode is again eligible for accounting

This occurs at the same time that all event data is purged as part of the Create Accounting program cycle.

- When the administrator launches the Purge Transaction Objects Diagnostics program

The table below describes the parameters for the Purge Transaction Objects Diagnostics program.

**Purge Transaction Objects Diagnostics Program Parameters**

---

| <b>Parameter</b>              | <b>Description</b>  |
|-------------------------------|---|
| Application                   | Restricts the purge of diagnostic data to a particular application; value defaulted to the application owning the responsibility from which the program is launched   |
| End Date                      | Purges the diagnostic data up to the given date   |
| Accounting Program Request ID | Purges the diagnostic data for a given request ID, which corresponds to an execution of the Create Accounting program<br><br>The list of values includes all request IDs for previous executions of the Create Accounting program with diagnostics enabled. |

---

---

## Subledger Journal Entries

### Subledger Journal Entries Overview

Create subledger journal entries as follows:

- Manually create subledger journal entries and assign supporting references at the header and line level in the Create Subledger Journal Entry page.

See: Creating Manual Subledger Journal Entries, page 6-2

- Use the Create Accounting program to process eligible accounting events to create subledger journal entries.

See: Overview of Create Accounting and Transfer Journal Entries to GL Programs, page 3-1

- Use the Manual Subledger Journal Entries API to create, update, delete, complete, and reverse subledger journal entries.

A manual subledger journal entry is not associated with a subledger transaction. When the manual subledger journal entry is complete, Subledger Accounting assigns the appropriate sequence names, sequence numbers, and creates the reporting currency journal entries if appropriate.

Enter manual subledger journal entries only against a primary or secondary ledger. Manual journal entries created for the primary ledgers are automatically posted to the reporting currency and not to the secondary ledger.

See:

- Subledger Journal Entries Inquiry, page 7-4
- Subledger Journal Entry Lines Inquiry, page 7-5

## Creating Manual Subledger Journal Entries

Use the Subledger Journal Entries - Search page to search for all types of subledger journal entries, not just manual entries. In the search results region:

- Drill down to view subledger journal entry details
- View, assign, and unassign supporting references
- Update, duplicate, complete, and reverse manual journal entries

The screenshot shows the Oracle Subledger Journal Entries - Create Journal Entry page. The top navigation bar includes 'Subledger Journal Entries', 'Accounting Events', and 'Supporting Reference Balances'. Below the navigation is a 'Create Subledger Journal Entry' button and three action buttons: 'Cancel', 'Save as Incomplete', and 'Continue'. A note indicates that an asterisk (\*) indicates a required field.

**Header**

|                |        |                        |                              |  |
|----------------|--------|------------------------|------------------------------|--|
| Balance Type   | Actual | * GL Date              | Assign Supporting References |  |
| * Ledger       |        | (example: 22-Aug-2007) |                              |  |
| Legal Entity   |        | * Category             |                              |  |
| Reference Date |        | * Description          |                              |  |

**Lines**

**TIP** Select the Default Currency Options button to change the default currency for new subledger journal entry lines.

| Details           | Number | *Account | *Accounting Class | Entered | Entered | Entered | Accounted | Accounted | Assign | Supporting | References | Delete |
|-------------------|--------|----------|-------------------|---------|---------|---------|-----------|-----------|--------|------------|------------|--------|
|                   |        |          |                   | DR      | CR      | DR      | CR        | CR        |        |            |            |        |
| No results found. |        |          |                   |         |         |         |           |           |        |            |            |        |

**Accounted Amounts**

Select the Calculate Totals to calculate the accounted amounts for all subledger journal entry lines.

Total Accounted Debit **0**      Total Accounted Credit **0**

**TIP** Select Save as Incomplete only if you want to save the journal entry as Incomplete and return to the Subledger Journal Entries page. Otherwise, select Continue.

**Subledger Journal Entries** | **Accounting Events** | **Supporting Reference Balances** | **Close Window** | **Preferences** | **Help** | **Diagnostics**  
Copyright (c) 2006, Oracle. All rights reserved.  
[About this Page](#)      [Privacy Statement](#)

## To Create Subledger Journal Entries

In the Subledger Journal Entries page, click **Create Journal Entry**.

### **Selected Fields in the Create Journal Entry Page**

| Field                        | Description  |
|------------------------------|--|
| Ledger                       | <p>The primary and secondary ledgers displayed in the ledger list of values are determined by the system profile option SLA: Enable Data Access Security in Subledgers. If this profile option is set to Yes, the list of values displays only the primary and secondary ledgers defined in the GL: Data Access Set and the SLA: Additional Data Access Set profile options. If the profile option is set to No, the list of values displays all primary and secondary ledgers.</p> <p>See:</p> <ul style="list-style-type: none"><li>• SLA: Enable Data Access Security in Subledgers, page B-8</li><li>• GL Profile Options, <i>Oracle General Ledger User Guide</i></li></ul> <p>If Encumbrance is selected in the Balance Type field, the list of values only displays ledgers that are enabled for budgetary control.</p> |
| Category                     | Journal category   |
| GL Date                      | GL date for posting the subledger journal entry.   |
| Reference Date               | Reference date to be stored at the subledger journal entry header level. Used in reporting sequence for Actual journal entries.  |
| Assign Supporting References | Opens the Assign Supporting References - Lines page where you can assign and unassign supporting references.   |

### **To Assign Supporting References at the Header or Line Level**

Use the Assign Supporting References page to view or modify current supporting reference values and to assign new supporting references to the journal entry or line. This page is available in view-only mode if you navigate from a final journal entry and

the page title will be displayed as View Supporting References.

To assign supporting references to the journal entry in the Header region of the Create Subledger Journal Entry page:

- Click Assign Supporting References.
- Click Add 5 Lines.
- Select the supporting reference in the Name field and enter details as required.
- Click Apply to save the changes.

**Note:** Supporting references with line level source assignments on the details may not be assigned on the journal entry header.

To assign supporting references to the lines of a journal entry in the Lines region of the Create Subledger Journal Entry page:

- Click the Assign Supporting References icon on the appropriate line
- Click Add 5 Lines.
- Select the supporting reference in the Name field and enter details as required.
- Click Apply to save the changes.

**Note:** Supporting references maintaining balances can only be assigned on journal entry lines.

## To Create Subledger Journal Lines

In the Lines region of the Create Subledger Journal Entry page:

- Enter account, debit and credit amounts, and accounting class.
- Optionally enter third party information, description, and currency details for each line of the subledger journal entry.

If third party control accounts are enabled for the application and the account entered is a third party control account, third party information must be entered. For example, if Payables is set up to support only Supplier, you cannot select third party control accounts that have been set up to support only Customer.

See: Third Party Balances Report, page 9-33

- View the total accounted amounts of all subledger journal entry lines in the ledger currency.

**Selected Fields and Buttons in the Create Subledger Journal Entry Page, Lines Region**

---

| Field and Button         | Description   |
|--------------------------|---|
| Default Currency Options | Opens the Default Currency Options page to default an entered currency, conversion type, conversion date, and conversion rate information for each new subledger journal entry line. Use the currency controls available in the Details region of each journal entry line to override the default set.  |
| Third Party Type         | Defaults from the application associated with the responsibility used to access the manual subledger journal entry pages. The values for name, site, number, and taxpayer ID must match the third party type. These values are required if the account on the line is a third party control account. If the account is not a third party control account, but party type is selected, party information should be provided. |
| Conversion Rate          | If the conversion type is User, you must enter a conversion rate.   |
| Calculate Totals         | Updates the display in the Accounted Amounts region   |
| Continue                 | Select to review the manual subledger journal entry and complete it.  |

---

## Enhanced Third Party Control Account Validations

The Third Party Control Account feature in the Oracle E-Business Suite enables organizations to maintain detailed balances by third party for an account code combination. Subledger Accounting checks that valid third party information is associated with the journal line if the account code combination is a Third Party Control Account. If it is not, then Subledger Accounting rejects the journal line. General Ledger prevents manual journal entries from posting to the third party control accounts. This ensures that journal lines that post to the control account codes such as Supplier Liability Account, Customer Receivables Account, are associated with the valid third party information in the respective subledger modules.

## Restricting Natural Account Values in the General Ledger Manual Journal Entries

You might sometimes define an account code as a third party control account to prevent the GL manual journal entries from posting to these accounts. When such accounts are used in the subledger transaction without the third party information, Subledger Accounting prevents the accounting for such transaction with an error that the third party information is missing.

For example, when the unidentified receipts account code in the Oracle Receivables is defined as a third party control account, accounting for the unidentified receipts fails because no third party information is available for such receipts. Similar issues are observed with the rounding account and intercompany accounts.

The following solution helps overcome these issues. A new third party control account validation is provided that allows accounting to complete successfully without the third party information in the subledgers but to restrict that account code while entering in the GL manual journal entries.

A new value, Restrict Manual Journals, is added to the List of values of the Third Party Control Account segment qualifier.

You can assign Restrict Manual Journals value in the list of values of the Third Party Control Account segment qualifier to natural account segment values and prevent the GL manual journal entries from posting to these account code combinations containing the respective natural account segment values.

When you define the Unidentified Receipt account code as third party control account to prevent the GL manual journal entries from using this account code, you define the Unidentified Receipt account code as Restrict Manual Journals third party control account. When you use this account code for the unidentified receipt in the Oracle Receivables, Subledger Accounting creates the accounting successfully for this unidentified receipt without the party information. Hence, you cannot enter manual journal entries using this account code in General Ledger.

After you set up an account code as Restrict Manual Journals third party control account, you must run the concurrent program, Program - Inherit Segment Value Attributes to reflect the Third Party Control Account qualifier changes in the account code combinations.

After this, you cannot select this account code while creating a manual journal entry in General Ledger. Though, you can select these account codes without the third party information in the respective subledgers as the application treats this third party control account for the purpose of restricting the usage in General Ledger manual journal entries only.

In General Ledger the validation performed on entering manual journals in journal entry window has been modified to restrict users from using the third party control accounts defined as Restrict Manual Journals.

Subledger Accounting is modified to treat the third party control accounts defined as Restrict Manual Journals the same way as the third party control accounts defined as No.

If an account or a control account is updated as Restrict Manual Journals or No from

either the Supplier, Customer, or Both, then the balances for the Third Party Balances Report is not maintained for such account code with the prospective effect. The balances prior to the modification will only be shown in the report output and thus the balance in the report output may not match with the General Ledger balance for such account or control code.

## Supporting Control Balance Initialization

You can upload initial balance value for the prior period activities and synchronize them. The initial balance of a Third Party Control Account is defined in context of ledger and is the starting balance for which there is no journal entry in Subledger Accounting to support that balance.

The Third Party Control Account feature of the Oracle Subledger Accounting (SLA) enables organizations to maintain the breakdown of an account balance by third party for asset and liability accounts.

When a new third party control account is created, the account does not have an opening balance and all the transactions posted to the account contain the third party information. SLA updates the third party balances for every transaction posted to the account. This ensures that user can generate the third party balances report and successfully reconcile the closing balance of the account with the sum of the third party balances for that account. When an existing account having balance is converted into a control account there will not be any third party balances in the Subledger Accounting and the third party balances report cannot be generated for that account.

This new feature lets you upload the initial third party balances for a control account and synchronize the account and third party balances for accurate reporting and reconciliation. A new open interface table (xla\_ctrl\_balances\_int) and an import program are provided to initialize the control account balances.

### Initialize Control Account Balances

You can upload initial balance for a third party control account for a combination of ledger, application, party and party-site combination. An initial balance is in the ledger currency for a particular period. This initial balance is carried forward to the subsequent periods.

You define initial balances as Net of Debit (Dr) and Credit (Cr) balance. You cannot define initial balance as separate balances for Debit and Credit.

You can define an initial balance for a closed period only. You can also define initial balance even after the SLA has created balances and entries for third party control accounts.

### Update or Delete Initial Balances

Once the initial balance has been entered, you can still modify the initial balance. Modifying the initial balance impacts the subsequent period balances. The following validations are performed:

- The periods prior to the date of initial balance must not have any period activity for the combination of third party control account, application, ledger, third party and third party site for which the initial balance is uploaded
- Initial balance is uploaded only for the third party control accounts.
- The period closing status of the periods for which the initial balance is being uploaded must be "Closed".

#### **Submit Concurrent Program to Import Data from Interface table into the Base table**

- Insert the data into the interface table (xla\_ctrl\_balances\_int)
- Run the concurrent program, Import Initial Balance for Third Party Control Account, from the responsibility where the data is defined and the application where the control balance data is intended. For example, submit the concurrent program from Payables responsibility

The only parameter for this concurrent program is the Purge Option. The valid values are Purge successfully processed rows, Purge all rows and Do not purge.

- If you specify Purge successfully processed rows, then only the successfully imported data in the interface table is deleted after processing.
- If you specify Purge all rows, then all data in the interface table is deleted after processing.
- If you specify Do not purge, then all the data in the interface table will be retained after processing.

The concurrent program once completed successfully generates a report.

The report is divided into the following sections:

- Successfully Initialized Rows
- Warnings - GL Balance does not match SLA Balance after initialization
- Failed Initialization Rows
- Error Code Details

#### **To Review and Complete Subledger Journal Entries**

In the Review Subledger Journal Entry page:

- Review all subledger journal entry header and line information
- View supporting references.

- If budgetary control is enabled for the primary ledger and the subledger journal entry fund status is Required and Failed, initiate funds check to reserve funds to complete the subledger journal entry. Subledger Accounting returns a status of Passed or Failed.

There is no manual reservation of funds and funds checking cannot be initiated for a budget journal entry that has a status of Final.

View the budgetary control validation results if the fund status is Failed by clicking the Funds Validation Results button.

- Complete the journal entry

The Complete button is enabled if the journal entry:

- Is a manual journal entry with a status of Draft, Error, or Incomplete
- Is an accrual reversal or multiperiod accounting recognition entry with an Incomplete status
- Does not belong to a reporting currency

#### ***Selected Subledger Journal Entry Completion Options***

---

| Option         | Description  |
|----------------|--|
| Draft          | Results cannot be transferred or posted to General Ledger; appears on Subledger Accounting reports or inquiries that are based upon draft entries; entries can be updated and deleted; and cannot perform funds reservation.   |
| Final          | Subledger journal entry is completed; can be transferred and posted to General Ledger; appears on Subledger Accounting reports and inquiries that are based upon final entries; and funds reservation is performed.  |
| Final and Post | Completes the subledger journal entry with the status Final; transfers it to General Ledger and posts it; entry appears on Subledger Accounting reports or inquiries that are based upon final entries. If the entry is successfully completed and transferred, the status is Transferred. |

---

Subledger Accounting completes the subledger journal entry as follows:

- Validates the accounts; same validation as that performed by the Create Accounting program  
See: Subledger Journal Entry Definition Overview, page C-1
- Assigns the corresponding sequence to the subledger journal entry if completed with Final status
- Updates the status of the subledger journal entry
- Updates the pending subledger journal entry with the corresponding error message if the subledger journal entry is not successfully created
- Invokes the balance update program

**Note:** If the subledger journal entry is for a ledger with reporting currencies, Subledger Accounting creates subledger journal entries for these representations.

## To Duplicate Subledger Journal Entries

In the Duplicate Subledger Journal Entries page, create a new subledger journal entry from an existing subledger journal entry of any status except Error. When a journal entry is duplicated, you can update all fields except Balance Type.

**Note:** If changing the ledger in a duplicated journal entry and the new ledger has a different chart of accounts, all lines are cleared because the Accounting Flexfield structure is dependent on the new ledger.

## To Reverse Subledger Journal Entries

In the Reverse Subledger Journal Entry page, reverse an existing manual subledger journal entry in Final status. Reversal options are:

- Switch DR/CR
- Change Sign

## To Update Subledger Journal Entries

In the Update Subledger Journal Entries page, update:

- A manual subledger journal entry of status Draft, Incomplete, or Error
- Supporting references for manual subledger journal entries with a status of Final

- A multiperiod recognition or accrual reversal journal entry of status Incomplete

When a manual journal entry is updated, you can update all fields except Balance Type and Ledger.

## **To Delete Subledger Journal Entries**

In the Delete Subledger Journal Entries page, delete a manual subledger journal entry for a primary or secondary ledger in Draft or Incomplete status.



---

## Inquiries

### Subledger Accounting Inquiries Overview

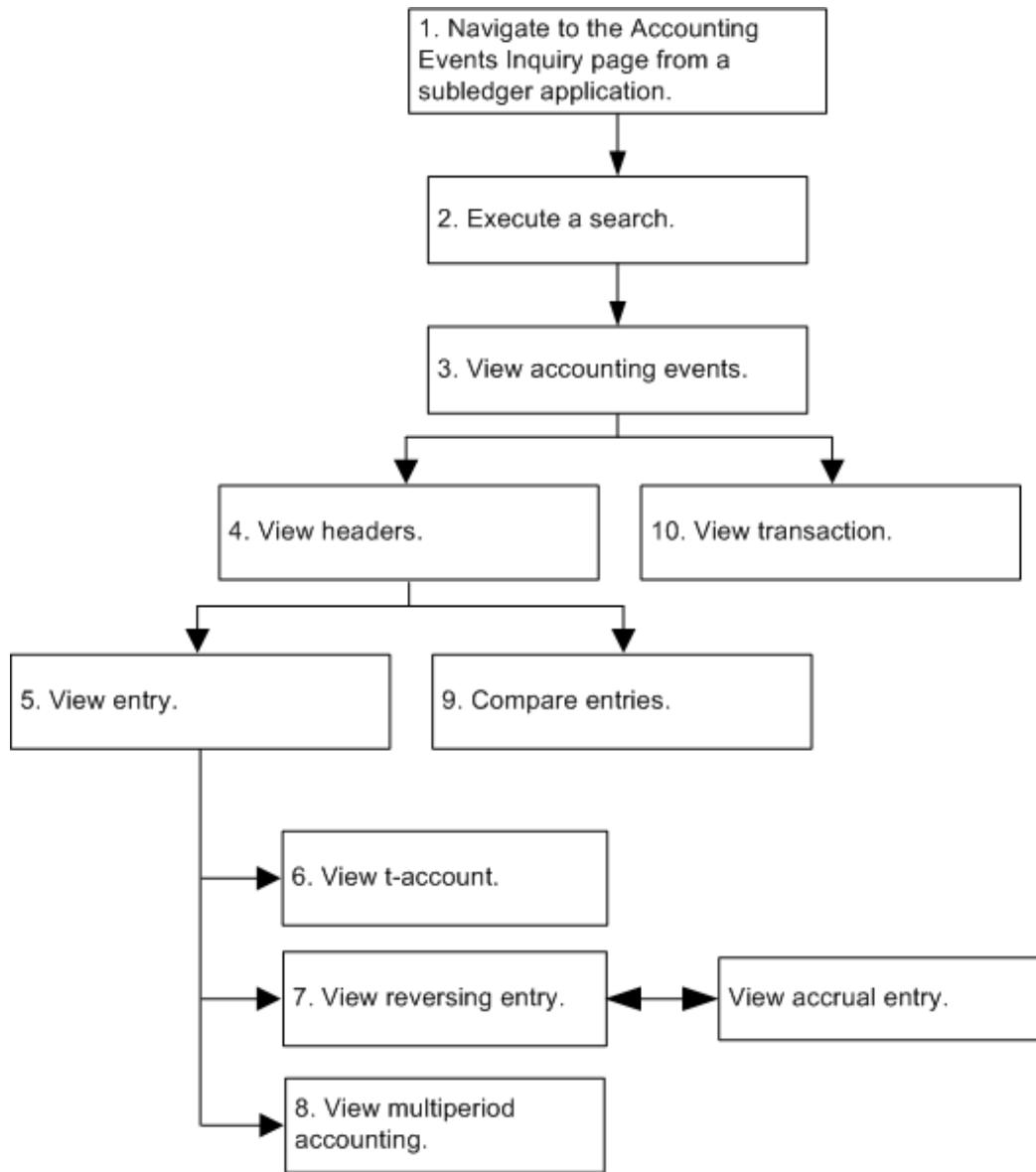
Use the accounting events drill-down and inquiries features to:

- Perform an inquiry on accounting events, journal entries and journal entry lines based on multiple selection criteria
- View information about an accounting event or journal entry error
- View detailed information about the subledger journal entry headers for a given accounting event
- Compare subledger journal entry information for any two journal entries
- View subledger journal entry lines for a number of different documents or transactions
- View subledger journal entry in t-account format
- View transactions underlying the accounting event or journal entry
- View supporting references associated with the subledger journal line.

### Accounting Events Inquiry

#### Accounting Events Inquiry Process Flow Diagram

The figure below shows the process flow for accounting events inquiry from a subledger application responsibility and is described in Accounting Events Inquiry Process Steps, page 7-2.



## Accounting Events Inquiry Process Steps

1. In the subledger application responsibility for any application using Subledger Accounting, navigate to the Accounting Events inquiry page.
2. Execute a search for specific accounting events.
3. View accounting events.
4. View subledger journal header information for a selected accounting event by clicking **View Journal Headers**.

**Note:** The View Subledger Journal Entry Headers page is displayed only if there is more than one subledger journal entry created for an accounting event. If there is only one subledger journal entry for an accounting event, the View Subledger Journal Entry page is displayed.

5. View detailed entry information about the subledger journal entry header by clicking the ledger link.

**Note:** The following conditions exist:

- If the balance type is Actual, the reference date is displayed.
- If the balance type is Budget, the Budget Version field is displayed.
- If the balance type is Encumbrance, the Encumbrance Type field is displayed.
- If the entry is for public sector and funds approval is required, the Funds Status field is displayed.

6. View journal entry information in a t-account format by clicking **View T-Account** in the Header region of the View Subledger Journal Entry page.
7. If viewing an accrual reversal journal entry in the View Subledger Journal Entry page, the **View Accrual Entry** button is displayed to view the accrual journal entry details. If viewing an accrual journal entry, the **View Reversing Entry** button is displayed to view the accrual reversal journal entry details.
8. View multiperiod lines associated with the accrual journal entry by clicking the Multiperiod icon in the Lines region.
9. Compare subledger journal entry header and line information for two selected journals by clicking **Compare** in the View Subledger Journal Entry Headers page.
10. View the transaction associated with the accounting event by clicking **View Transaction** in the View Subledger Journal Entry Headers page.

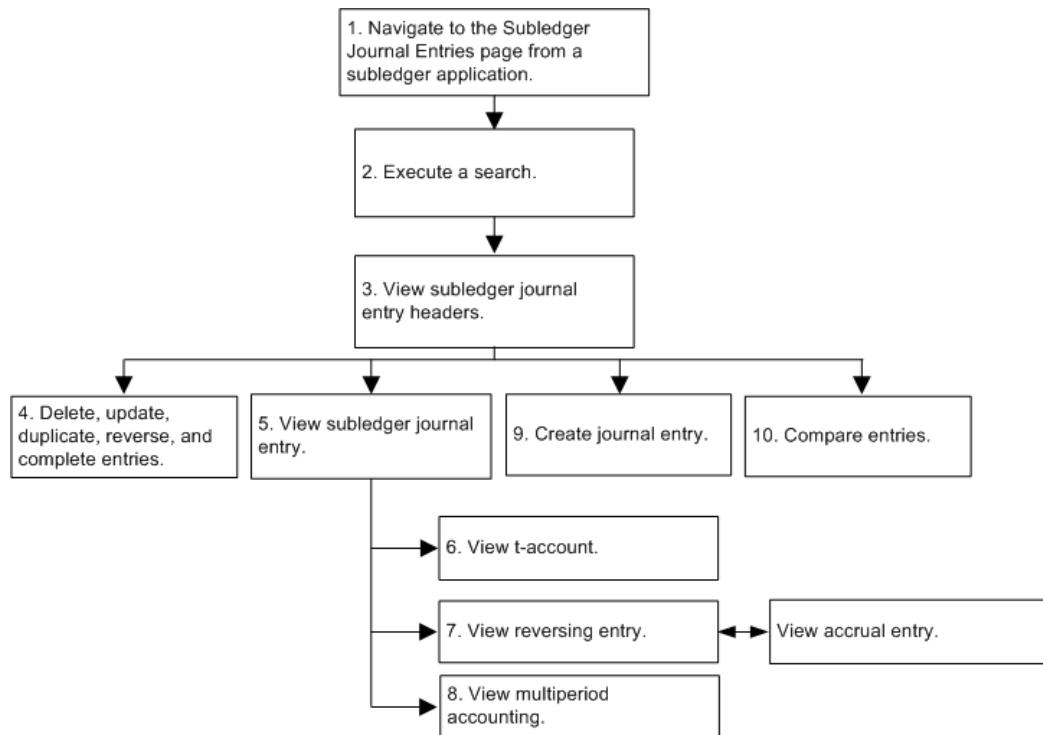
**Note:** This button appears only if the drilldown procedure is registered for the subledger application.
11. View supporting references at the header or line level for the subledger journal entry by clicking the Supporting References icon.

## Subledger Journal Entry Headers Inquiry

### Journal Entry Headers Inquiry Process Flow Diagram

The figure below shows the process flow for subledger journal entry headers inquiry and is described in Journal Entry Headers Inquiry Process Steps, page 7-4.

See: Subledger Journal Entry Definition Overview, page C-1



### Journal Entry Headers Inquiry Process Steps

1. In the subledger application responsibility for any application using Subledger accounting, navigate to the Subledger Journal Entry Headers inquiry page.
2. Execute a search.
3. View subledger journal entry headers in the results region.
4. Delete, update, duplicate, reverse, or complete subledger journal entries.
5. View detailed header and line information by selecting the ledger link.

See: Creating Manual Subledger Journal Entries, page 6-2

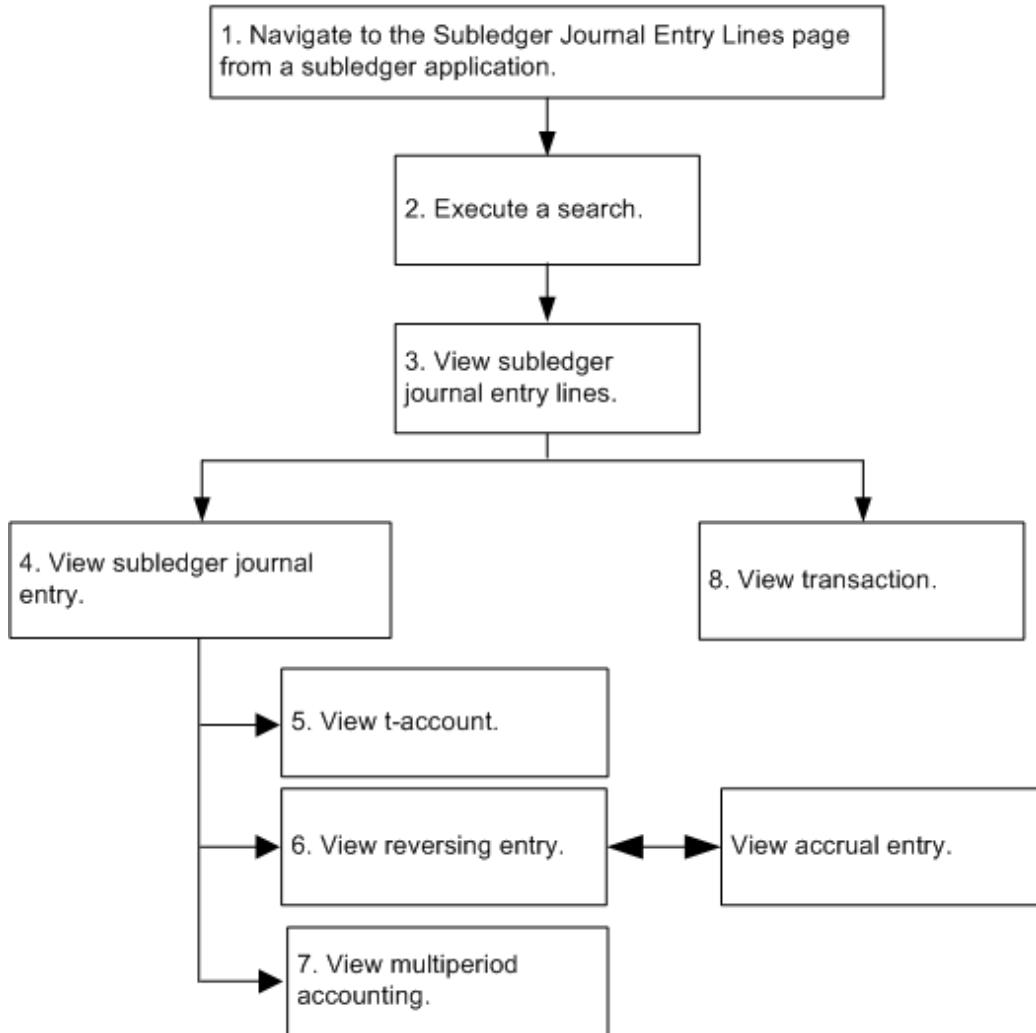
6. View journal entry information in a t-account format by clicking **View T-account** in the View Subledger Journal Entry page.
7. If viewing an accrual reversal journal entry in the View Subledger Journal Entry page, the **View Accrual Entry** button is displayed to view the accrual journal entry. If viewing an accrual journal entry, the **View Reversing Entry** button is displayed to view the accrual reversal journal entry.
8. View multiperiod lines associated with the accrual journal entry by clicking the Multiperiod icon in the Lines region.
9. Create journal entries page by clicking **Create Journal Entry** in the Subledger Journal Entries page.
10. Compare subledger journal entries by clicking **Compare** in the Subledger Journal Entries page.

## Subledger Journal Entry Lines Inquiry

View subledger journal entry lines for a number of different documents or transactions by selecting the Subledger Journal Entry Lines menu option in the subledger accounting responsibility. Use the Search region in the Subledger Journal Entry Lines page to execute an inquiry for specific events based on selected criteria.

## Journal Entry Lines Inquiry Process Flow Diagram

The figure below shows the process flow for journal entry lines inquiry from a subledger application and is described in Journal Entry Lines Inquiry Process Steps, page 7-6.



## Journal Entry Lines Inquiry Process Steps

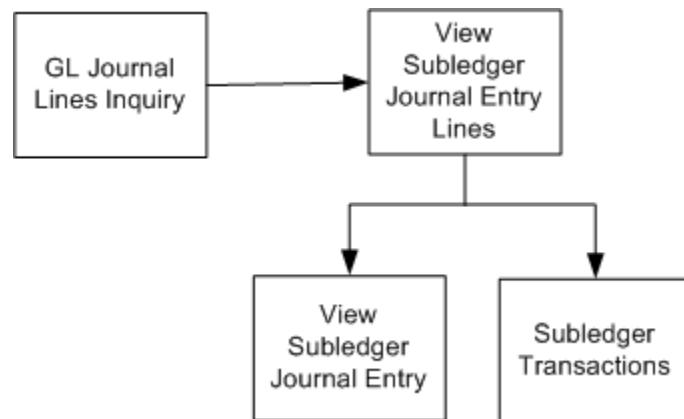
1. In the subledger application responsibility for any application using Subledger Accounting, navigate to the Subledger Journal Entry Lines page.
2. Execute a search.
3. View subledger journal entry lines in the results region.
4. View subledger journal entry lines and header for a selected journal entry line by clicking **View Journal Entry**.
5. View journal line information in t-account format by clicking **View T-account** in the View Subledger Journal Entry page.

6. Click **View Supporting References** to view the supporting references associated with the subledger journal line.
7. If viewing an accrual reversal journal entry in the View Subledger Journal Entry page, the **View Accrual Entry** button is displayed to view the accrual journal entry details. If viewing an accrual journal entry, the **View Reversing Entry** button is displayed to view the accrual reversal journal entry details.
8. View multiperiod lines associated with the accrual journal entry line by clicking the **Multiperiod** icon in the Lines region.
9. View the transaction associated with the subledger journal entry by clicking **View Transaction** in the Subledger Journal Entry Lines page.

**Note:** This button appears only if the drilldown procedure is registered for the subledger application.

## Drilldown from General Ledger

The figure below shows the drilldown from GL journal lines inquiry to subledger journal entry lines in Subledger Accounting. From here, users can navigate to the subledger journal entries or drill down to the subledger transaction.



## Viewing Supporting Reference Balances

Use the Supporting Reference Balances page to inquire on account balances for a particular supporting reference, then drill-down to the journal lines that contribute to the balances.

To view account balances for one or more supporting references:

1. Navigate to the Supporting Reference Balances page.

2. Enter required information: ledger, period range, and supporting reference name. Optionally, enter an account and supporting reference detail values.
3. Click Go.
4. Click on the appropriate period activity to view the journal entry lines.
5. On the View Journal Entry Lines page, click View Journal Entry to drill-down to the Subledger Journal Entry page and view the journal entry to which the line belongs or click View Transaction to drill-down to the subledger transaction that originates the journal entry.
6. Click the Supporting References icon to open the View Supporting References page.
7. On the Supporting Reference Balances page, click Export to download the balances to a spreadsheet.

---

## Reporting Sequence

### Reporting Sequence Overview

Journal entries have the following sequences attached to Subledger Accounting:

- accounting sequence

The accounting sequence is assigned to Subledger Accounting journal entries at the time that the journal entry is completed. The sequence is assigned in the completion date order using the GL date as the date criterion for determining the sequence to be used.

- reporting sequence

The reporting sequence is assigned to both Subledger Accounting journal entries and General Ledger journal entries when the General Ledger 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.

These two sequences are not mutually exclusive and can coexist in the same journal entry.

In some cases, the completed journal entries can be functionally correct but the sequence numbers assigned to them are wrong. For example, the user realizes that the sequencing setup is not correct and wants to resequence all the journal entries that are sequenced based on the wrong setup. Subledger Accounting provides an undo and redo accounting feature to correct the original accounting. A key component of this feature is the Re-sequencing of Journal Entries.

Reporting sequence is optional.

## Sequence Programs

The reporting sequence feature is based on the following programs:

- Reporting Sequence program

This program belongs to Subledger Accounting and selects the journal entries to be sequenced. This program is called in the following modes:

- Reporting sequence functionality
- Undo sequences functionality
- Accounting Sequence Numbering program

The reporting sequence program selects the journal entries to be sequenced and performs the following:

- Groups journal entries in a specific order
- Sequences the journal entries that are completed and transferred to General Ledger
- Assigns sequence numbers that are consecutive with no gaps to journal entries that share the same sequence
- Assigns reporting sequence at the ledger level providing the following combinations for multiple ledgers:
  - A different sequence is assigned to each ledger.
  - Different reporting sequences are assigned to some ledgers, and there is no sequence assignment to the rest of the ledgers.
- Sequences journal entries that belong to an adjustment period when the corresponding nonadjustment period, based on the sequence control date, is closed

**Note:** If a customer plans to use reporting sequence for a ledger, and the ledger calendar includes adjustment periods, then adjustment periods must:

- Always coincide with the last day of another nonadjustment period
- Cover only one day
- Resequences journal entries

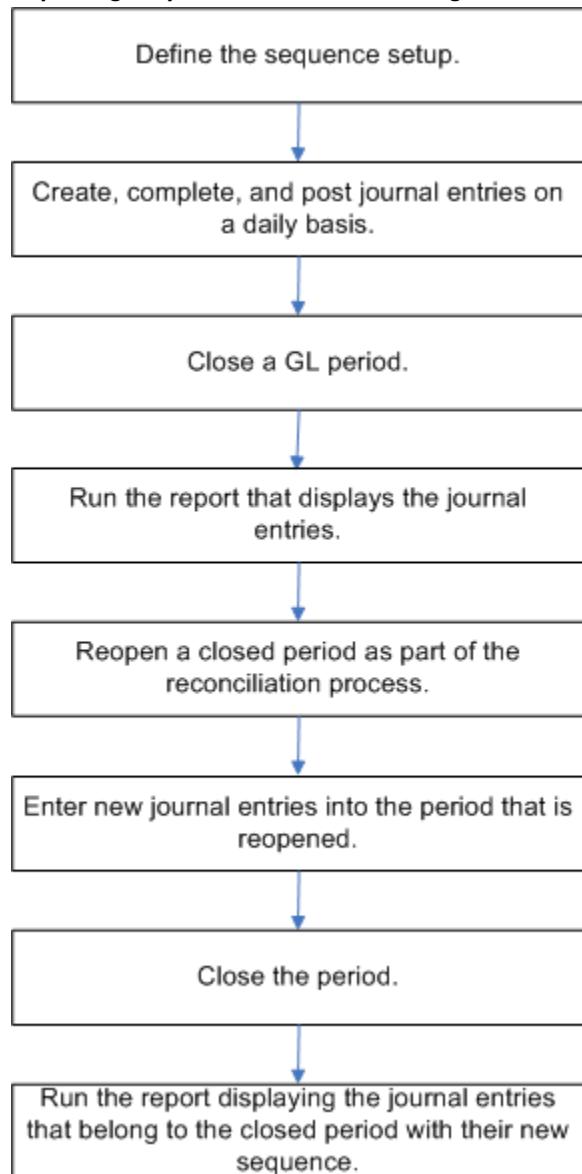
## Resequence of Accounting Sequence Number Program

This program is for use by Oracle Development and Support to maintain the reporting order sequence when journal entries are deleted or added.

## Reporting Sequence Process Flow Diagram

The diagram below shows the reporting sequence process and is described in Reporting Sequence Process Steps, page 8-4.

***Reporting Sequence Process Flow Diagram***



## Reporting Sequence Process Steps

1. Define the sequence setup as part of the implementation process.
2. Create, complete, and post journal entries as part of the daily operations.
3. Close a General Ledger period as part of the period end operations and the reporting sequence program is automatically run.

After selecting the journal entries that belong to this period and to all the subsequent consecutive closed periods, the reporting sequence program calls the accounting sequence numbering program to perform the actual sequencing. The reporting sequence program selects the journal entries in chronological order based on the sequencing order.

- When a GL period is closed, the journal entries are sequenced if all the prior GL periods are closed based on the sequence control date.

When there is one previous GL period open, the journal entries that belong to a closed period are not eligible for sequencing.

4. Run the report that displays the journal entries.

Run this report whenever you want. The journal entries that belong to a closed period appear in this report with their corresponding sequence number, while the journal entries that belong to an open period are displayed without the sequence number since they have not been sequenced yet.

5. You can decide to open a closed period as part of the reconciliation process and the reporting sequence program is automatically run.

The program erases the sequences that are assigned to the journal entries that belong to the reopened period. If there are other succeeding closed periods, the sequences assigned to the journal entries that belong to these periods are also erased. As the sequence assigned to the journal entries is a gapless sequence, the reporting sequence program calls the accounting sequence numbering program in erase mode with the lowest sequence assigned to the selected journal entries to avoid gaps when the journal entries are resequenced.

6. Enter new journal entries into the period that has been reopened.

7. Close the period and the reporting sequence program is automatically run.

The reporting sequence program calls the accounting sequence numbering program to sequence all the journal entries that belong to this period in chronological order based on the sequencing order. The reporting sequence program selects the journal entries that belong to the succeeding closed periods.

The following journal entries can be assigned a reporting sequence:

- In Subledger Accounting, journal entries that are complete and transferred to General Ledger

**Note:** Draft entries are not assigned a reporting sequence.

- In General Ledger, posted journal entries

8. Run the report that displays all the journal entries that belong to the closed periods with their new sequence.



---

# Subledger Accounting Reports

## Subledger Accounting Reports Overview

Oracle Subledger Accounting provides accounting reports for fiscal and internal control purposes. Create your own layouts and publish your reports using Oracle XML Publisher.

XML Publisher is a template-based publishing tool that is delivered with the Oracle E-Business suite that enables users to rapidly develop and maintain custom report formats. You can design and control how the reports are presented by using report templates. When generating a report, XML Publisher merges report template files with report data to create documents that support numerous formatting options, including colors, images, font styles, and headers and footers.

See Introduction, *Oracle XML Publisher User's Guide*

Subledger Accounting provides the following reports:

- Journal Entries Report, page 9-3
- Account Analysis Report, page 9-20
- Third Party Balances Report, page 9-33
- Multiperiod Accounting Reports, page 9-47
- Period Close Exceptions Report, page 9-62
- Subledger Accounting Rules Detail Listing Report, page 9-66

This chapter describes the parameters and data elements for each report.

System administrators can define new concurrent programs using any subset of the parameters. Users can create new XML Publisher templates using any subset of the data elements.

See: Overview of Concurrent Programs and Requests, *Oracle Applications System Administrator's Guide*, Creating a Template, *Oracle XML Publisher User's Guide*

## XML Output File

When generating a report using a report template, first generate an XML output file of the requested report. XML Publisher applies the requested report template to the XML output file of the requested report to generate a file in the requested format.

The figure below shows a sample XML output file.

```
<?xml version="1.0" ?>
- <!--
Generated by Oracle Reports version 6.0.8.22.0
-->
- <GLRTBD>
  - <G_PAGEBREAK>
    - <G_DETAIL>
      <BEGIN_BALANCE>-388650</BEGIN_BALANCE>
      <ACTIVITY>0</ACTIVITY>
      <END_BALANCE>-388650</END_BALANCE>
      <ACCT_VALUE>2925</ACCT_VALUE>
      <ACCT_VALUE_DESC>Revenue Clearing</ACCT_VALUE_DESC>
      <ACCT>01-600-2925-0000-700</ACCT>
    </G_DETAIL>
    <PAGEBREAK_NAME>Product</PAGEBREAK_NAME>
    <PAGEBREAK_VALUE>700</PAGEBREAK_VALUE>
    <ACTIVITY_TOTAL>0</ACTIVITY_TOTAL>
    <BEGIN_BALANCE_TOTAL>-388650</BEGIN_BALANCE_TOTAL>
    <END_BALANCE_TOTAL>-388650</END_BALANCE_TOTAL>
    <PAGEBREAK_VALUE_DESC>Direct Finance Lease</PAGEBREAK_VALUE_DESC>
    <PAGEBREAK_ROW_COUNT>1</PAGEBREAK_ROW_COUNT>
  </G_PAGEBREAK>
  <LEDGER_NAME_PARAM>Vision Operations (USA)</LEDGER_NAME_PARAM>
  <ACCOUNT_NAME_COLUMN>Acct</ACCOUNT_NAME_COLUMN>
  <PAGEBREAK_LEFTPROMPT_DSP>Product</PAGEBREAK_LEFTPROMPT_DSP>
  <ROW_COUNT>1</ROW_COUNT>
  <PERIOD_NAME_PARAM>Feb-03</PERIOD_NAME_PARAM>
  <CURRENCY_PARAM>USD</CURRENCY_PARAM>
  <PAGEBREAK_VALUE_FROM_PARAM>000</PAGEBREAK_VALUE_FROM_PARAM>
  <PAGEBREAK_VALUE_TO_PARAM>999</PAGEBREAK_VALUE_TO_PARAM>
  <AMOUNT_TYPE_PARAM>Period to Date</AMOUNT_TYPE_PARAM>
</GLRTBD>
```

XML output files are composed of tag set elements. For example, in the figure above, <BEGIN\_BALANCE>-388650</BEGIN\_BALANCE> is the beginning balance element, which consists of a starting and ending tag. The data between the tags is the value of the element. In the figure above, the value for the beginning balance element is -388650.

## Parameters

The parameters in some of the following reports are categorized as follows:

- Selection parameters: filter the information retrieved by the report

- Custom parameters: used to pass information to the template, such as a boilerplate text or formatting options

The custom parameters described in the table below apply to all the Subledger Accounting reports.

***Custom Parameters***

| Parameter          | Token                    | Required | Valid Values  |
|--------------------|--------------------------|----------|---------------|
| Custom Parameter 1 | P_CUSTOM_PARA<br>METER_1 | No       | No validation |
| Custom Parameter 2 | P_CUSTOM_PARA<br>METER_2 | No       | No validation |
| Custom Parameter 3 | P_CUSTOM_PARA<br>METER_3 | No       | No validation |
| Custom Parameter 4 | P_CUSTOM_PARA<br>METER_4 | No       | No validation |
| Custom Parameter 5 | P_CUSTOM_PARA<br>METER_5 | No       | No validation |

## Journal Entries Report

The Journal Entries Report displays detailed header and line information for subledger and General Ledger journal entries

**Note:** To avoid duplication with subledger journal entries, General Ledger journal entries imported from Subledger Accounting are not included in the report.

## Journal Entries Report Parameters

### *Journal Entries Report Selection Parameters*

| Parameter                | Token                  | Required | Valid Values   |
|--------------------------|------------------------|----------|--|
| Ledger /Ledger Set       | P_LEDGER               | Yes      | From GL: All ledgers and ledger sets included in the access set identified by the profile option SLA: Data Access Set<br><br>From Subledger, Use Ledger Security = Yes: All ledgers and ledger sets included in the access sets identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set<br><br>From Subledger, Use Ledger Security = No: All ledgers and ledger sets defined in GL |
| Legal Entity             | P_LEGAL_ENTITY         | No       | All legal entities assigned to ledgers in the ledger set   |
| Transaction Legal Entity | P_TRX_LEGAL_ENTI<br>TY | No       | All legal entities in the business group   |
| Period [From, To]        |                        | Yes      |  |
| From                     | P_PERIOD_FROM          |          | All periods from the accounting calendar associated with the selected Ledger or Ledger Set   |
| To                       | P_PERIOD_TO            |          | All periods from the accounting calendar equal to or coming after the From period up to the end of the fiscal year   |
| GL Date [From, To]       |                        | No       |  |

| Parameter                   | Token                   | Required | Valid Values  |
|-----------------------------|-------------------------|----------|---|
| From                        | P_GL_DATE_FROM          |          |   |
| To                          | P_GL_DATE_TO            |          | Any date later than or equal to the From date   |
| Creation Date [From, To]    |                         | No       |   |
| From                        | P_CREATION_DATE_FROM    |          |   |
| To                          | P_CREATION_DATE_TO      |          | Any date later than or equal to the From date   |
| Transaction Date [From, To] |                         | No       |   |
| From                        | P_TRANSACTION_DATE_FROM |          |   |
| To                          | P_TRANSACTION_DATE_TO   |          | Any date later than or equal to the From date   |
| Journal Entry Status        | P_JE_STATUS             | No       | Valid values are Draft, Final, and Error.   |
| Posted                      | P_POSTING_STATUS        | No       | Yes: Lines that are posted in GL (i.e. transferred to and posted in GL)<br><br>No: Lines that are not posted in GL (not transferred to GL or transferred to GL and not posted)<br><br>All: all lines, irrespective of their posting status. |
| Journal Source              | P_JE_SOURCE             | No       | From GL: All GL journal sources<br><br>From Subledger: taken from the application owning the responsibility   |

| Parameter                             | Token                              | Required  | Valid Values  |
|---------------------------------------|------------------------------------|---|---|
| Process Category                      | P_PROCESS_CATEGORY                 | No; requires a source associated with a subledger application | All process categories defined for the application associated with the journal entry source |
| Accounting Event Class                | P_EVENT_CLASS                      | No; requires a source associated with a subledger application | All event classes enabled for the application associated with the JE source                 |
| Accounting Sequence Name              | P_ACCOUNTING_SEQUENCE_NAME         | No  | All sequences assigned to ledgers in the ledger or ledger set                               |
| Accounting Sequence Version           | P_ACCOUNTING_SEQUENCE_VERSION      | No; requires accounting sequence name                         | All sequence versions for the accounting sequence   |
| Accounting Sequence Number [From, To] |                                    | No; requires accounting sequence name and version             |   |
| From                                  | P_ACCOUNTING_SEQUENCE_NUMBE_R_FROM |   |   |
| To                                    | P_ACCOUNTING_SEQUENCE_NUMBE_R_TO   |   | Any value greater than or equal to the From value   |
| Reporting Sequence Name               | P_REPORTING_SEQUENCE_NAME          | No  | All available sequences assigned to ledgers in the ledger or ledger set                     |

| Parameter                            | Token                                    | Required   | Valid Values   |
|--------------------------------------|--|--|--|
| Reporting Sequence Version           | P_REPORTING_SEQ<br>UENCE_VERSION         | No; requires reporting sequence name             | All sequence versions for the reporting sequence         |
| Reporting Sequence Number [From, To] |  | No; requires reporting sequence name and version |  |
| From                                 | P_REPORTING_SEQ<br>UENCE_NUMBER_F<br>ROM |  |  |
| To                                   | P_REPORTING_SEQ<br>UENCE_NUMBER_T<br>O   |  | Any value greater than or equal to the From value        |
| Document Sequence Name               | P_DOCUMENT_SEQ<br>UENCE_NAME             | No   | All available document sequences                         |
| Document Sequence Number [From, To]  |  | No; requires document sequence name              |  |
| From                                 | P_DOCUMENT_SEQ<br>UENCE_NUMBER_F<br>ROM  |  |  |
| To                                   | P_DOCUMENT_SEQ<br>UENCE_NUMBER_T<br>O    |  | Any value greater than or equal to the From value        |
| Third Party Type                     | P_PARTY_TYPE                             | No   | Customer or Supplier                                     |
| Third Party Name                     | P_PARTY_NAME                             | No; requires third party type                    | All customers or suppliers based on the third party type |

| Parameter                            | Token                           | Required                            | Valid Values   |
|--------------------------------------|---------------------------------|-------------------------------------|--|
| Third Party Number<br>[From, To]     |                                 | No; requires<br>third party<br>type |  |
| From                                 | P_PARTY_NUMBER<br>_FROM         |                                     | All customer or supplier<br>numbers based on the third<br>party type   |
| To                                   | P_PARTY_NUMBER<br>_TO           |                                     | Any customer or supplier<br>number greater than or equal<br>to the From value  |
| Journal Category                     | P_JE_CATEGORY                   | No                                  | All available journal<br>categories  |
| Balance Type                         | P_BALANCE_TYPE                  | No                                  | Valid values are Actual,<br>Encumbrance, and Budget.   |
| Budget                               | P_BUDGET_NAME                   | No                                  | All enabled budgets  |
| Encumbrance Type                     | P_ENCUMBRANCE_<br>TYPE          | No                                  | All encumbrance types<br>defined in GL   |
| Include Zero Amount<br>Lines         | P_INCLUDE_ZERO_<br>AMOUNT_LINES | No                                  | Yes and No; controls whether<br>lines with both entered and<br>accounted amount equal to<br>zero are included in the<br>report |
| Entered Currency                     | P_ENTERED_CURRE<br>NCY          | No                                  | All enabled currencies   |
| Side                                 | P_SIDE                          | No                                  | Debit or credit. A journal<br>entry line is included in the<br>extract only if the line has the<br>side provided by the user.  |
| Accounted Amount<br>Range [From, To] |                                 | No                                  | Allowed only if a ledger<br>(rather than a ledger set) is<br>provided  |
| From                                 | P_ACCOUNTED_A<br>MOUNT_FROM     |                                     |  |

| Parameter                          | Token                     | Required | Valid Values   |
|------------------------------------|---------------------------|----------|--|
| To                                 | P_ACCOUNTED_A<br>MOUNT_TO |          | Any value greater than or equal to the From date   |
| Balancing Segment [From, To]       |                           | No       | From GL: All balancing segment values (BSV) included in the access set identified by the profile option GL: Data Access Set  |
|                                    |                           |          | From Subledger, Use Ledger Security = Yes: All BSVs assigned to the ledger or ledger set (parameter) and included in the access sets identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set |
|                                    |                           |          | From Subledger, Use Ledger Security = No: All BSVs assigned to the ledger or ledger set  |
| From                               | P_BALANCING_SEGMENT_FROM  |          |  |
| To                                 | P_BALANCING_SEGMENT_TO    |          | All BSVs greater than or equal to the From value that meet the above criteria  |
| Natural Account Segment [From, To] |                           | No       |  |
| From                               | P_ACCOUNT_SEGMENT_FROM    |          | All enabled natural account values from the ledger or ledger set's chart of accounts   |
| To                                 | P_ACCOUNT_SEGMENT_TO      |          | Any enabled natural account value from the ledger or ledger set's chart of accounts greater than or equal to the From value  |

| Parameter                              | Token                            | Required  | Valid Values  |
|--|----------------------------------|---|---|
| Accounting Code Combination [From, To] |                                  | No  |   |
| From                                   | P_ACCOUNTING_F<br>LEXFIELD _FROM |   | All enabled accounting code combinations from the ledger or ledger set's chart of accounts  |
| To                                     | P_ACCOUNTING_F<br>LEXFIELD_TO    |   | Any enabled accounting code combination from the ledger or Ledger Set's COA greater than or equal to the From value   |
| Valuation Method                       | P_VALUATION_ME<br>THOD           | No  | No validation   |
| Security Id Integer 1                  | P_SECURITY_ID_IN<br>T_1          | No  | No validation   |
| Security Id Integer 2                  | P_SECURITY_ID_IN<br>T_2          | No  | No validation   |
| Security Id Integer 3                  | P_SECURITY_ID_IN<br>T_3          | No  | No validation   |
| Security Id Char 1                     | P_SECURITY_ID_CH<br>AR_1         | No  | No validation   |
| Security Id Char 2                     | P_SECURITY_ID_CH<br>AR_2         | No  | No validation   |
| Security Id Char 3                     | P_SECURITY_ID_CH<br>AR_3         | No  | No validation   |
| Post-Accounting Program                | P_POST_ACCOUNTING_PROGRAM        | No; requires a source associated with a subledger application | All post-accounting programs defined for the subledger application associated with the journal entry source. The extract retrieves lines with accounting classes assigned to the post-accounting program. |

| Parameter                            | Token                          | Required                            | Valid Values   |
|--------------------------------------|--------------------------------|-------------------------------------|--|
| User Transaction Identifier Column 1 | P_USER_TRX_ID_COL_1 (to 5)     | No; requires an event class         | All columns from the transaction view associated with the event class  |
| User Transaction Identifier Value 1  | P_USER_TRX_ID_VALUE_1 (to 5)   | No; requires an event class         | No validation  |
| GL Batch Name                        | P_GL_BATCH_NAME                | No; requires a journal entry source | All GL batches from the journal entry source   |
| Include User Transaction Identifiers | P_INCLUDE_USER_TRX_IDENTIFIERS | No                                  | Yes or No; controls whether the extract program retrieves user identifiers names and values  |
| Include Tax Details                  | P_INCLUDE_TAX_DETAILS          | No                                  | Yes or No; controls whether the extract program retrieves tax details for the journal entry lines  |
| Include Legal Entity Information     | P_INCLUDE_LEGAL_ENTITY_INFO    | No                                  | Legal Entity: Company information is retrieved from the legal entity to which the balancing segment value of the line is assigned; Establishment: Company information is retrieved from the establishment to which the balancing segment value of the line is assigned; None: No company information is retrieved. |

## Journal Entries Report Data Elements

**Note:** Not all data elements have a corresponding field on the seeded journal entry report templates.

### **Journal Entries Report Data Elements**

| <b>Field Name</b>      | <b>Data Element (XML Tag)</b> | <b>Description</b>                               |
|------------------------|-------------------------------|--|
| GL Date                | GL_DATE                       | Journal entry GL date                            |
| Transaction Created by | CREATED_BY                    | User who created the subledger transaction       |
| Creation Date          | CREATION_DATE                 | Journal entry creation date                      |
| Last Update Date       | LAST_UPDATE_DATE              | Last date journal entry updated                  |
| GL Transfer Date       | GL_TRANSFER_DATE              | Date journal entry transferred to General Ledger |
| Reference Date         | REFERENCE_DATE                | Journal entry legal reporting date               |
| Completed Date         | COMPLETED_DATE                | Journal entry completion date                    |
| External Reference     | EXTERNAL_REFERENCE            | Journal entry external reference                 |
| Period Year            | PERIOD_YEAR                   | Journal entry fiscal year                        |
| Period Year Start Date | PERIOD_YEAR_START_DATE        | First date of the fiscal year                    |
| Period Year End Date   | PERIOD_YEAR_END_DATE          | Last date of the fiscal year                     |
| Period Number          | PERIOD_NUMBER                 | Position of the period in the fiscal year        |
| Period Name            | PERIOD_NAME                   | Journal entry accounting period name             |
| Period Start Date      | PERIOD_START_DATE             | First date of accounting period                  |

| Field Name                  | Data Element (XML Tag)      | Description  |
|-----------------------------|-----------------------------|--|
| Period End Date             | PERIOD_END_DATE             | Last date of accounting period   |
| Transaction Number          | TRANSACTION_NUMBER          | Subledger transaction number   |
| Transaction Date            | TRANSACTION_DATE            | Subledger transaction date   |
| Accounting Sequence Name    | ACCOUNTING_SEQUENCE_NAME    | Sequence name used to number the journal entry at creation or posting    |
| Accounting Sequence Version | ACCOUNTING_SEQUENCE_VERSION | Sequence version used to number the journal entry at creation or posting |
| Accounting Sequence Number  | ACCOUNTING_SEQUENCE_NUMBER  | Sequence number assigned to the journal entry at creation or posting     |
| Reporting Sequence Name     | REPORTING_SEQUENCE_NAME     | Sequence name used to number the journal entry at period closing         |
| Reporting Sequence Version  | REPORTING_SEQUENCE_VERSION  | Sequence version used to number the journal entry at period closing      |
| Reporting Sequence Number   | REPORTING_SEQUENCE_NUMBER   | Sequence number assigned to the journal entry at period closing          |
| Document Sequence Category  | DOCUMENT_CATEGORY           | Document category assigned to the subledger transaction                  |
| Document Sequence Name      | DOCUMENT_SEQUENCE_NAME      | Sequence name used to number the subledger transaction                   |
| Document Sequence Number    | DOCUMENT_SEQUENCE_NUMBER    | Sequence number assigned to the subledger transaction                    |

| Field Name            | Data Element (XML Tag) | Description   |
|-----------------------|------------------------|---|
| Application Id        | APPLICATION_ID         | Internal identifier of the application owning the journal entry                     |
| Application Name      | APPLICATION_NAME       | Application name owning the journal entry   |
| Ledger ID             | LEDGER_ID              | Internal identifier of the ledger owning the journal entry                          |
| Ledger Short Name     | LEDGER_SHORT_NAME      | Short name of ledger owning the journal entry                                       |
| Ledger Description    | LEDGER_DESCRIPTION     | Description of ledger owning the journal entry                                      |
| Ledger Name           | LEDGER_NAME            | Name of ledger owning the journal entry   |
| Ledger Currency       | LEDGER_CURRENCY        | Currency of ledger owning the journal entry   |
| Header ID             | HEADER_ID              | Journal entry internal identifier   |
| Header Description    | HEADER_DESCRIPTION     | Journal entry description   |
| Journal Entry Status  | JOURNAL_ENTRY_STATUS   | Subledger journal entry status: Draft, Final, and Enter                             |
| Transfer to GL Status | TRANSFER_TO_GL_STATUS  | Indicates whether journal entry transferred to General Ledger                       |
| Balance Type Code     | BALANCE_TYPE_CODE      | Balance type indicator: A, B, or E for Actual, Budget, and Encumbrance respectively |

| Field Name            | Data Element (XML Tag) | Description  |
|-----------------------|------------------------|--|
| Balance Type Name     | BALANCE_TYPE           | Journal entry balance type: Actual, Balance, or Encumbrance                |
| Budget Name           | BUDGET_NAME            | Budget name for budget entries   |
| Encumbrance Type      | ENCUMBRANCE_TYPE       | Encumbrance type for encumbrance entries                                   |
| Fund Status Value     | FUND_STATUS            | Fund reservation status for the journal entry                              |
| Journal Category Name | JE_CATEGORY_NAME       | Journal entry category   |
| Journal Source Name   | JE_SOURCE_NAME         | Journal entry source   |
| Event ID              | EVENT_ID               | Internal identifier of accounting event that originated the journal entry  |
| Event Date            | EVENT_DATE             | Date of the accounting event that originated the journal entry             |
| Event Number          | EVENT_NUMBER           | Number of the accounting event that originated the journal entry           |
| Event Class Code      | EVENT_CLASS_CODE       | Event class code of the accounting event that originated the journal entry |
| Event Class Name      | EVENT_CLASS_NAME       | Event class name of the accounting event that originated the journal entry |
| Event Type Code       | EVENT_TYPE_CODE        | Event type code of the accounting event that originated the journal entry  |

| Field Name                        | Data Element (XML Tag)                  | Description  |
|-----------------------------------|---|--|
| Event Type Name                   | EVENT_TYPE_NAME                         | Event type name of the accounting event that originated the journal entry                                    |
| User Transaction Identifier Name  | USER_TRX_IDENTIFIER_NAME<br>E_1 (to 10) | Additional transaction information for identifying the transaction   |
| User Transaction Identifier Value | USER_TRX_IDENTIFIER_VAL<br>UE_1 (to 10) | Additional transaction information for identifying the transaction   |
| Batch Name                        | GL_BATCH_NAME                           | Name of General Ledger batch that includes the journal entry   |
| GL Batch Status                   | GL_BATCH_STATUS                         | Status of batch that includes the journal entry: Unposted or Posted  |
| Posted Date                       | POSTED_DATE                             | Date journal entry posted to General Ledger  |
| GL Journal Name                   | GL_JE_NAME                              | Journal entry name; name for subledger journal entries taken from corresponding General Ledger journal entry |
| GL Document Sequence Name         | GL_DOC_SEQUENCE_NAME                    | General Ledger document sequence name  |
| GL Document Sequence Value        | GL_DOC_SEQUENCE_VALUE                   | Document sequence value in General Ledger  |
| GL Line Number                    | GL_LINE_NUMBER                          | Sequence name used to number the journal in General Ledger at creation                                       |
| Line Number                       | LINE_NUMBER                             | Sequence number assigned to the journal in General Ledger at creation  |

| Field Name                   | Data Element (XML Tag)                                   | Description   |
|------------------------------|--|---|
| Accounting Class Code        | ACCOUNTING_CLASS_CODE                                    | Accounting class code assigned to the subledger journal entry line                      |
| Accounting Class             | ACCOUNTING_CLASS_NAME                                    | Accounting class assigned to the subledger journal entry line                           |
| Line Description             | LINE_DESCRIPTION   | Journal entry line description  |
| Code Combination Id          | CODE_COMBINATION_ID                                      | Accounting code combination internal identifier for the journal entry line              |
| Accounting Code Combination  | ACCOUNTING_CODE_COMBINATION (concatenated values)        | Concatenated values of the accounting code combination for the journal entry line       |
| Code Combination Description | CODE_COMBINATION_DESCRIPTION (concatenated descriptions) | Concatenated descriptions of the accounting code combination for the journal entry line |
| Control Account Flag         | CONTROL_ACCOUNT_FLAG                                     | Indicates whether the account is a third party control identifier: Y for yes; N for No  |
| Control Account              | CONTROL_ACCOUNT  | Translated value of the control account flag: Yes or No                                 |
| Entered Currency             | ENTERED_CURRENCY   | Journal entry currency  |
| Conversion Rate              | CONVERSION_RATE  | Transaction conversion rate   |
| Conversion Rate Date         | CONVERSION_RATE_DATE                                     | Transaction conversion rate date  |

| Field Name                 | Data Element (XML Tag)    | Description   |
|----------------------------|---------------------------|---|
| Conversion Rate Type Code  | CONVERSION_RATE_TYPE_CODE | Transaction conversion type code of rate                            |
| Conversion Rate Type       | CONVERSION_RATE_TYPE      | Transaction conversion rate type                                    |
| Entered Debit              | ENTERED_DR                | Entered debit for journal entry line                                |
| Entered Credit             | ENTERED_CR                | Entered credit for journal entry line                               |
| Unrounded Accounted Debit  | UNROUNDED_ACCOUNTED_DR    | Accounted debit before rounding                                     |
| Unrounded Accounted Credit | UNROUNDED_ACCOUNTED_CR    | Accounted credit before rounding                                    |
| Accounted Debit            | ACCOUNTED_DR              | Journal entry line accounted debit                                  |
| Accounted Credit           | ACCOUNTED_CR              | Journal entry line accounted credit                                 |
| Statistical Amount         | STATISTICAL_AMOUNT        | Journal entry line statistical amount                               |
| Reconciliation Reference   | RECONCILIATION_REFERENCE  | Reference used for reconciliation purposes                          |
| Attribute Category         | ATTRIBUTE_CATEGORY        | Context column for the journal entry lines descriptive flexfield    |
| Attribute 1 (to 10)        | ATTRIBUTE1 (to 10)        | Attribute columns for the journal entry lines descriptive flexfield |
| Third Party Type           | PARTY_TYPE_CODE           | Third party type indicator: C for Customer; S for Supplier          |

| Field Name                               | Data Element (XML Tag)             | Description   |
|--|------------------------------------|---|
| Third Party Type                         | PARTY_TYPE                         | Party type: Customer or Supplier                            |
| Third Party Number                       | PARTY_NUMBER                       | Third party number for journal entry line                   |
| Third Party Name                         | PARTY_NAME                         | Third party name for journal entry line                     |
| Third Party Taxpayer Id                  | PARTY_TYPE_TAXPAYER_ID             | Third party taxpayer identifier for journal entry line      |
| Third Party Tax Registration Number      | PARTY_TAX_REGISTRATION_NUMBER      | Third party tax registration number for journal entry line  |
| Third Party Site Number                  | PARTY_SITE_NUMBER                  | Third party site number for journal entry line              |
| Third Party Site Name                    | PARTY_SITE_NAME                    | Third party site name for journal entry line                |
| Third Party Tax Registration Number      | PARTY_TAX_REGISTRATION_NUMBER      | Third party tax registration number for journal entry line  |
| Third Party Site Number                  | PARTY_SITE_NUMBER                  | Third party site number for journal entry line              |
| Third Party Site Name                    | PARTY_SITE_NAME                    | Third party site name for journal entry line                |
| Third Party Site Tax Registration Number | PARTY_SITE_TAX_REGISTRATION_NUMBER | Third party site taxpayer identifier for journal entry line |
| Balancing Segment                        | BALANCING_SEGMENT                  | Journal entry line balancing segment                        |
| Balancing Segment Description            | BALANCING_SEGMENT_DESCRIPTION      | Journal entry line balancing segment description            |

| Field Name                        | Data Element (XML Tag)           | Description  |
|-----------------------------------|----------------------------------|--|
| Natural Account                   | NATURAL_ACCOUNT                  | Journal entry line natural account segment value                   |
| Natural Account Description       | NATURAL_ACCOUNT_DESCRIPTION      | Journal entry line natural account segment description             |
| Cost Center                       | COST_CENTER                      | Journal entry line cost center segment value                       |
| Cost Center Description           | COST_CENTER_DESCRIPTION          | Journal entry line cost center segment description                 |
| Management Segment                | MANAGEMENT_SEGMENT               | Journal entry line management segment value                        |
| Management Segment Description    | MANAGEMENT_SEGMENT_DESCRIPTION   | Journal entry line management segment description                  |
| Inter-company Segment             | INTERCOMPANY_SEGMENT             | Journal entry line intercompany segment value                      |
| Inter-company Segment Description | INTERCOMPANY_SEGMENT_DESCRIPTION | Journal entry line intercompany segment description                |
| Segment 1 (to 30)                 | SEGMENT1 (to 30)                 | Journal entry lines individual Accounting Flexfield segment values |

## Account Analysis Report

The 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 transferred to and posted to General Ledger.

**Note:** To avoid duplication with subledger journal entries, General Ledger journal entries imported from Subledger Accounting are not included in the report.

## Account Analysis Report Parameters

### *Account Analysis Report Selection Parameters*

| Parameter          | Token          | Required | Valid Values   |
|--------------------|----------------|----------|--|
| Ledger /Ledger Set | P_LEDGER       | Yes      | All ledgers and ledger sets included in the access set identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set   |
| Legal Entity       | P_LEGAL_ENTITY | No       | All legal entities assigned to ledgers in the ledger set   |
| Period [From, To]  |                | Yes      |  |
| From               | P_PERIOD_FROM  |          | All periods for the accounting calendar associated with the selected Ledger or Ledger Set  |
| To                 | P_PERIOD_TO    |          | All periods for the accounting calendar equal to or after the From period up to the end of the fiscal year   |
| GL Date [From, To] |                | No       | Allows users to specify a General Ledger range inside the period range. Journal entry lines with GL dates between the start date of the Period From and the GL Date From are summarized. Similarly, Journal entries with GL dates between the GL Date To and the end date of the Period To are summarized. |
| From               | P_GL_DATE_FROM |          | Any date in the Period From  |

| Parameter                    | Token              | Required | Valid Values   |
|------------------------------|--------------------|----------|--|
| To                           | P_GL_DATE_TO       |          | Any date in the Period To, later than or equal to the From date  |
| Balance Type                 | P_BALANCE_TYPE     | No       | <ul style="list-style-type: none"> <li>• Actual</li> <li>• Encumbrance</li> <li>• Budget</li> </ul>  |
| Budget                       | P_BUDGET_NAME      | No       | All enabled budgets for ledgers in the ledger set  |
| Encumbrance Type             | P_ENCUMBRANCE_TYPE | No       | All encumbrance types defined in General Ledger  |
| Balance Side                 | P_BALANCE_SIDE     | No       | <ul style="list-style-type: none"> <li>• Debit</li> <li>• Credit</li> </ul>  |
|                              |                    |          | A balance is included in the report only if the net period ending balance (period beginning balance + net period activity) has the side provided by the user.  |
| Balance Amount<br>[From, To] |                    | No       | Allowed only if a ledger is provided. Detail for an accounting code combination and period is included in the report only if the period ending balance (period beginning balance + period net activity) is within the amount range. Used in conjunction with Balance Side. |
| From                         | P_AMOUNT_FROM      |          |  |
| To                           | P_AMOUNT_TO        |          | Any value greater than or equal to the From value  |

| Parameter                          | Token                       | Required | Valid Values  |
|------------------------------------|-----------------------------|----------|---|
| Balancing Segment<br>[From, To]    |                             | No       |   |
| From                               | P_BALANCING_SEGMENT_FROM    |          | All balancing segment values (BSV) included in the access set identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set |
| To                                 | P_BALANCING_SEGMENT_TO      |          | All BSVs greater than or equal to the From value that meet the above criteria   |
| Natural Account Segment [From, To] |                             | No       |   |
| From                               | P_ACCOUNT_SEGMENT_FROM      |          | All enabled natural account values from the ledger or ledger set's chart of accounts  |
| To                                 | P_ACCOUNT_SEGMENT_TO        |          | Any enabled natural account value from the ledger or ledger set's chart of accounts greater than or equal to the From value                             |
| Accounting Flexfield [From, To]    |                             | No       |   |
| From                               | P_ACCOUNTING_FLEXFIELD_FROM | Yes      | All enabled accounting code combinations from the ledger or ledger set's chart of accounts  |
| To                                 | P_ACCOUNTING_FLEXFIELD_TO   | Yes      | Any enabled accounting code combination from the ledger or ledger set's chart of accounts greater than or equal to the From value                       |

| Parameter                            | Token                          | Required | Valid Values   |
|--------------------------------------|--------------------------------|----------|--|
| Include Zero Amount Balances         | P_INCLUDE_ZERO_AMOUNT_BALANCES | No       | Yes or No; controls whether accounting code combinations with period beginning balance equal to zero and no period activity are included in the report   |
| Include Zero Amount Lines            | P_INCLUDE_ZERO_AMOUNT_LINES    | No       | Yes or No; controls whether journal entry lines with both entered and accounted amounts equal to zero are included in the report   |
| Include User Transaction Identifiers | P_INCLUDE_USER_TRX_IDENTIFIERS | No       | Yes or No; controls whether the report program retrieves user transaction identifiers names and values   |
| Include Tax Details                  | P_INCLUDE_TAX_DETAILS          | No       | Yes or No; controls whether the report program retrieves tax details for the journal entry lines   |
| Include Legal Entity Information     | P_INCLUDE_LEGAL_ENTITY_INFO    | No       | Legal Entity: Company information is retrieved from the legal entity to which the balancing segment value of the line is assigned; Establishment: Company information is retrieved from the establishment to which the balancing segment value of the line is assigned; None: No company information is retrieved. |

## Account Analysis Data Elements

**Note:** Not all data elements have a corresponding field on the seeded account analysis report templates.

**Account Analysis Report Ledger and General Ledger Balance Information Data Elements**

| Field Name         | Data Element (XML Tag) | Descriptions  |
|--------------------|------------------------|---|
| Ledger ID          | LEDGER_ID              | Internal identifier of ledger owning the journal entry                              |
| Ledger Short Name  | LEDGER_SHORT_NAME      | Short name of ledger owning the journal entry                                       |
| Ledger Description | LEDGER_DESCRIPTION     | Description of ledger owning the journal entry                                      |
| Ledger Name        | LEDGER_NAME            | Name of ledger owning the journal entry   |
| Ledger Currency    | LEDGER_CURRENCY        | Currency of ledger owning the journal entry   |
| Period Year        | PERIOD_YEAR            | Journal entry fiscal year   |
| Period Number      | PERIOD_NUMBER          | Position of the period in the fiscal year   |
| Period Name        | PERIOD_NAME            | Journal entry accounting period name  |
| Period Start Date  | PERIOD_START_DATE      | First date of accounting period   |
| Period End Date    | PERIOD_END_DATE        | Period end date   |
| Balance Type Code  | BALANCE_TYPE_CODE      | Balance type indicator: A, B, or E for Actual, Budget, and Encumbrance respectively |
| Balance Type Name  | BALANCE_TYPE           | Journal entry balance type: Actual, Balance, or Encumbrance                         |
| Budget Name        | BUDGET_NAME            | Budget name for budget entries  |

| Field Name   | Data Element (XML Tag)                                  | Descriptions   |
|--|---|--|
| Encumbrance Type   | ENCUMBRANCE_TYPE  | Encumbrance type for encumbrance entries   |
| Beginning Period Balance Dr                              | BEGIN_BALANCE_DR  | Beginning period debit balance   |
| Beginning Period Balance Cr                              | BEGIN_BALANCE_CR  | Beginning period credit balance  |
| Period Activity Dr                                       | PERIOD_NET_DR   | Period activity, debit   |
| Period Activity Cr                                       | PERIOD_NET_CR   | Period activity, credit  |
| Code Combination Id                                      | CODE_COMBINATION_ID                                     | Accounting code combination internal identifier for the journal entry line             |
| Accounting Code Combination                              | ACCOUNTING_CODE_COMBINATION (concatenated descriptions) | Concatenated values of the accounting code combination for the journal entry line      |
| Code Combination Description (concatenated descriptions) | CODE_COMBINATION_DESCRIPTION                            | Code combination description   |
| Control Account Flag                                     | CONTROL_ACCOUNT_FLAG                                    | Indicates whether the account is a third party control identifier: Y for Yes; N for No |
| Control Account Indicator                                | CONTROL_ACCOUNT   | Translated value of the control account flag: Yes or No                                |
| Balancing Segment  | BALANCING_SEGMENT                                       | Journal entry line balancing segment   |
| Balancing Segment Description                            | BALANCING_SEGMENT_DESCRIPTION                           | Journal entry line balancing segment description                                       |
| Natural Account  | NATURAL_ACCOUNT   | Journal entry line natural account segment value                                       |

| Field Name  | Data Element (XML Tag)           | Descriptions   |
|---|----------------------------------|--|
| Natural Account Description   | NATURAL_ACCOUNT_DESCRIPTION      | Journal entry line natural account description                                   |
| Cost Center   | COST_CENTER                      | Journal entry line cost center segment value                                     |
| Cost Center Description   | COST_CENTER_DESCRIPTION          | Journal entry line cost center segment description                               |
| Management Segment  | MANAGEMENT_SEGMENT               | Journal entry line cost center segment description                               |
| Management Segment Description  | MANAGEMENT_SEGMENT_DESCRIPTION   | Journal entry line management segment description                                |
| Inter-company Segment   | INTERCOMPANY_SEGMENT             | Journal entry line intercompany segment value                                    |
| Inter-company Segment Description   | INTERCOMPANY_SEGMENT_DESCRIPTION | Journal entry line intercompany segment description                              |
| Segment 1 (to 30)   | SEGMENT1 (to 30)                 | Journal entry lines individual Accounting Flexfield segment values               |
| Begin Running Total Dr<br>(Populated only if user provides a GL Date range) | BEGIN_RUNNING_TOTAL_DR           | Debit running total from the start date of the first period to the GL Date From  |
| Begin Running Total Cr<br>(Populated only if user provides a GL Date range) | BEGIN_RUNNING_TOTAL_CR           | Credit running total from the start date of the first period to the GL Date From |
| End Running Total Dr<br>(Populated only if user provides a GL Date range)   | END_RUNNING_TOTAL_DR             | Debit running total from the GL Date To to the end date of the last period       |

| Field Name  | Data Element (XML Tag)   | Descriptions  |
|---|--------------------------|---|
| End Running Total Cr<br>(Populated only if user provides a GL Date range) | END_RUNNING_TOTAL_C<br>R | Credit running total from the GL Date To to the end date of the last period |

#### ***Account Analysis Report Journal Entry Information Data Elements***

| Field Name                  | Data Element (XML Tag)      | Description  |
|-----------------------------|-----------------------------|--|
| GL Date                     | GL_DATE                     | Journal entry GL date  |
| Transaction Created by      | CREATED_BY                  | User who created the subledger transaction                               |
| Creation Date               | CREATION_DATE               | Journal entry creation date  |
| Last Update Date            | LAST_UPDATE_DATE            | Last date journal entry updated  |
| GL Transfer Date            | GL_TRANSFER_DATE            | Date journal entry transferred to General Ledger                         |
| Reference Date              | REFERENCE_DATE              | Journal entry legal reporting date                                       |
| Completed Date              | COMPLETED_DATE              | Journal entry completion date  |
| Transaction Number          | TRANSACTION_NUMBER          | Subledger transaction number   |
| Transaction Date            | TRANSACTION_DATE            | Subledger transaction date   |
| Accounting Sequence Name    | ACCOUNTING_SEQUENCE_NAME    | Sequence name used to number the journal entry at creation or posting    |
| Accounting Sequence Version | ACCOUNTING_SEQUENCE_VERSION | Sequence version used to number the journal entry at creation or posting |

| Field Name                 | Data Element (XML Tag)     | Description  |
|----------------------------|----------------------------|--|
| Accounting Sequence Number | ACCOUNTING_SEQUENCE_NUMBER | Sequence number assigned to the journal entry at creation or posting |
| Reporting Sequence Name    | REPORTING_SEQUENCE_NAME    | Sequence name used to number the journal entry at period closing     |
| Reporting Sequence Version | REPORTING_SEQUENCE_VERSION | Sequence version used to number the journal entry at period closing  |
| Reporting Sequence Number  | REPORTING_SEQUENCE_NUMBER  | Sequence number assigned to the journal entry at period closing      |
| Document Sequence Category | DOCUMENT_CATEGORY          | Document category assigned to the subledger transaction              |
| Document Sequence Name     | DOCUMENT_SEQUENCE_NAME     | Sequence name used to number the subledger transaction               |
| Document Sequence Number   | DOCUMENT_SEQUENCE_NUMBER   | Sequence number assigned to the subledger transaction                |
| Application Id             | APPLICATION_ID             | Internal identifier of the application owning the journal entry      |
| Application Name           | APPLICATION_NAME           | Application name owning the journal entry                            |
| Header Id                  | HEADER_ID                  | Journal entry internal identifier                                    |
| Header Description         | HEADER_DESCRIPTION         | Journal entry description  |
| Fund Status Value          | FUND_STATUS                | Fund reservation status for the journal entry                        |
| Journal Category Name      | JE_CATEGORY_NAME           | Journal entry category   |

| Field Name                        | Data Element (XML Tag)                  | Description  |
|-----------------------------------|---|--|
| Journal Source Name               | JE_SOURCE_NAME                          | Journal entry source   |
| Event Id                          | EVENT_ID                                | Internal identifier of accounting event that originated the journal entry  |
| Event Date                        | EVENT_DATE                              | Date of the accounting event that originated the journal entry             |
| Event Number                      | EVENT_NUMBER                            | Number of the accounting event that originated the journal entry           |
| Event Class Code                  | EVENT_CLASS_CODE                        | Event class code of the accounting event that originated the journal entry |
| Event Class Name                  | EVENT_CLASS_NAME                        | Event class name of the accounting event that originated the journal entry |
| Event Type Code                   | EVENT_TYPE_CODE                         | Event type code of the accounting event that originated the journal entry  |
| Event Type Name                   | EVENT_TYPE_NAME                         | Event type name of the accounting event that originated the journal entry  |
| User Transaction Identifier Name  | USER_TRX_IDENTIFIER_NA<br>ME_1 (to 10)  | Additional transaction information for identifying the transaction         |
| User Transaction Identifier Value | USER_TRX_IDENTIFIER_VA<br>LUE_1 (to 10) | Additional transaction information for identifying the transaction         |
| Batch Name                        | GL_BATCH_NAME                           | General Ledger batch name that includes the journal entry                  |
| Posted Date                       | POSTED_DATE                             | Date journal entry posted to General Ledger                                |

| Field Name                | Data Element (XML Tag)    | Description  |
|---------------------------|---------------------------|--|
| GL Journal Name           | GL_JE_NAME                | Journal entry name; name for subledger journal entries taken from corresponding General Ledger journal entry |
| GL Line Number            | GL_LINE_NUMBER            | Sequence name used to number the journal in General Ledger at creation                                       |
| Line Number               | LINE_NUMBER               | Sequence number assigned to the journal in General Ledger at creation  |
| Accounting Class Code     | ACCOUNTING_CLASS_CODE     | Accounting class code assigned to the subledger journal entry line   |
| Accounting Class          | ACCOUNTING_CLASS_NAME     | Accounting class assigned to the subledger journal entry line  |
| Line Description          | LINE_DESCRIPTION          | Journal entry line description   |
| Entered Currency          | ENTERED_CURRENCY          | Journal entry currency   |
| Conversion Rate           | CONVERSION_RATE           | Transaction conversion rate  |
| Conversion Rate Date      | CONVERSION_RATE_DATE      | Transaction conversion rate date   |
| Conversion Rate Type Code | CONVERSION_RATE_TYPE_CODE | Transaction conversion rate type code  |
| Conversion Rate Type      | CONVERSION_RATE_TYPE      | Transaction conversion rate type   |
| Entered Debit             | ENTERED_DR                | Entered debit for journal entry line   |
| Entered Credit            | ENTERED_CR                | Entered credit for journal entry line  |

| Field Name                 | Data Element (XML Tag)   | Description   |
|----------------------------|--------------------------|---|
| Unrounded Accounted Debit  | UNROUNDED_ACCOUNTED_DR   | Accounted debit before rounding                                     |
| Unrounded Accounted Credit | UNROUNDED_ACCOUNTED_CR   | Accounted credit before rounding                                    |
| Accounted Debit            | ACCOUNTED_DR             | Journal entry line accounted debit                                  |
| Accounted Credit           | ACCOUNTED_CR             | Journal entry line accounted credit                                 |
| Statistical Amount         | STATISTICAL_AMOUNT       | Journal entry line statistical amount                               |
| Reconciliation Reference   | RECONCILIATION_REFERENCE | Reference used for reconciliation purposes                          |
| Attribute Category         | ATTRIBUTE_CATEGORY       | Context column for the journal entry lines descriptive flexfield    |
| Attribute 1 (to 10)        | ATTRIBUTE1 (to 10)       | Attribute columns for the journal entry lines descriptive flexfield |
| Third Party Type           | PARTY_TYPE_CODE          | Third party type indicator: C for Customer; S for Supplier          |
| Third Party Type           | PARTY_TYPE               | Party Type: Customer or Supplier                                    |
| Third Party Number         | PARTY_NUMBER             | Third party number for journal entry line                           |
| Third Party Name           | PARTY_NAME               | Third party name for journal entry line                             |
| Third Party Taxpayer Id    | PARTY_TAXPAYER_ID        | Third party taxpayer identifier for journal entry line              |

| Field Name                               | Data Element (XML Tag)             | Description  |
|--|------------------------------------|--|
| Third Party Tax Registration Number      | PARTY_TAX_REGISTRATION_NUMBER      | Third party tax registration number for journal entry line |
| Third Party Site Number                  | PARTY_SITE_NUMBER                  | Third party site number for journal entry line             |
| Third Party Site Name                    | PARTY_SITE_NAME                    | Third party site name for journal entry line               |
| Third Party Site Tax Registration Number | PARTY_SITE_TAX_REGISTRATION_NUMBER | Third party tax registration number for journal entry line |

## Third Party Balances Report

The Third Party Balances Report displays balance and account activity information for suppliers and customers. It retrieves the following information:

- Third party balances for 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 event

## Third Party Balances Report Parameters

### *Third Party Balances Report Selection Parameters*

| Parameter          | Token    | Required | Valid Values  |
|--------------------|----------|----------|---|
| Ledger /Ledger Set | P_LEDGER | Yes      | All ledgers and ledger sets included in the access sets identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set |

| Parameter            | Token          | Required | Valid Values  |
|----------------------|----------------|----------|---|
| Journal Entry Source | P_JE_SOURCE    | No       | All journal entry sources associated with subledger applications that use third party control accounts  |
|                      |                |          | Can be changed only if the SLA: Allow Reports Journal Source Override profile options is enabled  |
|                      |                |          | See: SLA: Allow Reports Journal Source Override, page B-8   |
| Legal Entity         | P_LEGAL_ENTITY | No       | All legal entities assigned to ledgers in the ledger set  |
| Period [From, To]    |                | Yes      |   |
| From                 | P_PERIOD_FROM  |          | All periods for the accounting calendar and period type associated with the selected ledger or ledger set   |
| To                   | P_PERIOD_TO    |          | All periods from the accounting calendar equal to or after the From period up to the end of the fiscal year   |
| GL Date [From, To]   |                | No       | Specifies a General Ledger range inside the period range. Journal entry lines with GL dates between the start date of the Period From and the GL Date From are summarized. Journal entries with GL dates between the GL Date To and the end date of the Period To are summarized. |
| From                 | P_GL_DATE_FROM |          | Any date in the Period From   |
| To                   | P_GL_DATE_TO   |          | Any date in the Period To, later than or equal to the From date   |

| Parameter                        | Token                 | Required                               | Valid Values  |
|----------------------------------|-----------------------|--|---|
| Balance Side                     | P_BALANCE_SIDE        | No                                     | <ul style="list-style-type: none"> <li>• Debit</li> <li>• Credit</li> </ul>   |
|                                  |                       |  | A balance is included in the report only if the net period ending balance (period beginning balance + period net activity) has the side provided by the user.   |
| Balance Amount<br>[From, To]     |                       | No;<br>requires<br>Balance<br>Side     | Allowed only if a ledger is provided. A balance is included in the report only if the period ending balance (period beginning balance + period net activity) is within the amount range; used in conjunction with Balance Side. |
| From                             | P_BALANCE_AMOUNT_FROM |  |   |
| To                               | P_BALANCE_AMOUNT_TO   |  | Any value greater than or equal to the From value   |
| Third Party Type                 | P_PARTY_TYPE          | No                                     | Values include Customer, Supplier, and All.   |
| Third Party Name                 | P_PARTY_NAME          | No                                     | All customers or all suppliers based on the third party type  |
| Third Party Site                 | P_PARTY_SITE          | No;<br>requires<br>Third Party<br>Name | All third party sites for the supplier or customer identified by third party name   |
| Third Party Number<br>[From, To] |                       | No                                     |   |
| From                             | P_PARTY_NUMBER_FROM   |  | All customer or all supplier numbers based on the third party type  |

| Parameter                          | Token                       | Required | Valid Values  |
|------------------------------------|-----------------------------|----------|---|
| To                                 | P_PARTY_NUMBER<br>_TO       |          | Any customer or supplier numbers greater than or equal to the From value  |
| Balancing Segment<br>[From, To]    |                             | No       |   |
| From                               | P_BALANCING_SEGMENT_FROM    |          | All balancing segment values (BSV) tied to legal entities assigned to ledgers in the access set identified by the profile options GL: Data Access Set and SLA: Additional Data Access Set |
| To                                 | P_BALANCING_SEGMENT_TO      |          | All BSVs greater than or equal to the From value that meet the above criteria   |
| Natural Account Segment [From, To] |                             | No       |   |
| From                               | P_ACCOUNT_SEGMENT_FROM      |          | All enabled natural account values from the ledger or ledger set's chart of accounts  |
| To                                 | P_ACCOUNT_SEGMENT_TO        |          | Any enabled natural account value from the ledger or ledger set's chart of accounts greater than or equal to the From value   |
| Accounting Flexfield<br>[From, To] |                             | No       |   |
| From                               | P_ACCOUNTING_FLEXFIELD_FROM | Yes      | All enabled accounting code combinations from the ledger or ledger set's chart of accounts  |
| To                                 | P_ACCOUNTING_FLEXFIELD_TO   | Yes      | Any enabled accounting code combination from the ledger or ledger set's chart of accounts greater than or equal to the From value   |

| Parameter                            | Token                          | Required | Valid Values   |
|--------------------------------------|--------------------------------|----------|--|
| Include Zero Amount Balances         | P_INCLUDE_ZERO_AMOUNT_BALANCES | No       | Yes or No; controls whether third party and third party sites with period beginning balance equal to zero and no period activity are included in the report  |
| Include User Transaction Identifiers | P_INCLUDE_USER_TRX_IDENTIFIERS | No       | Yes or No; controls whether the report retrieves user transaction identifiers names and values   |
| Include Tax Details                  | P_INCLUDE_TAX_DETAILS          | No       | Yes or No; controls whether the report retrieves tax details for the journal entry lines   |
| Include Legal Entity Information     | P_INCLUDE_LEGAL_ENTITY_INFO    | No       | Legal Entity: Company information is retrieved from the legal entity to which the balancing segment value of the line is assigned; Establishment: Company information is retrieved from the establishment to which the balancing segment value of the line is assigned; None: No company information is retrieved. |

## Third Party Balances Report Data Elements

**Note:** Not all data elements have a corresponding field on the seeded report templates.

### *Third Party Balances Report Data Elements*

| Field Name | Data Element (XML Tag) | Description  |
|------------|------------------------|--|
| Ledger Id  | LEDGER_ID              | Internal identifier of ledger owning the journal entry |

| Field Name                          | Data Element (XML Tag)        | Description  |
|-------------------------------------|-------------------------------|--|
| Ledger Short Name                   | LEDGER_SHORT_NAME             | Short name of ledger owning the journal entry              |
| Ledger Description                  | LEDGER_DESCRIPTION            | Description of ledger owning the journal entry             |
| Ledger Name                         | LEDGER_NAME                   | Name of ledger owning the journal entry                    |
| Ledger Currency                     | LEDGER_CURRENCY               | Currency of ledger owning the journal entry                |
| Third Party Type                    | PARTY_TYPE_CODE               | Third party type indicator: C for Customer; S for supplier |
| Third Party Type                    | PARTY_TYPE                    | Party type: Customer or Supplier                           |
| Third Party Id                      | PARTY_ID                      | Third party internal identifier for journal entry line     |
| Third Party Number                  | PARTY_NUMBER                  | Third party number for journal entry line                  |
| Third Party Name                    | PARTY_NAME                    | Third party name for journal entry line                    |
| Third Party Taxpayer Id             | PARTY_TYPE_TAXPAYER_ID        | Third party taxpayer identifier for journal entry line     |
| Third Party Tax Registration Number | PARTY_TAX_REGISTRATION_NUMBER | Third party tax registration number for journal entry line |
| Third Party Address Line 1          | PARTY_ADDRESS_1               | Third party address line 1                                 |
| Third Party Address Line 2          | PARTY_ADDRESS_2               | Third party address line 2                                 |
| Third Party Address Line 3          | PARTY_ADDRESS_3               | Third party address line 3                                 |
| Third Party Address Line 4          | PARTY_ADDRESS_4               | Third party address line 4                                 |

| Field Name                               | Data Element (XML Tag)             | Description   |
|--|------------------------------------|---|
| Third Party City                         | PARTY_CITY                         | Third party city  |
| Third Party Postal Code                  | PARTY_ZIP_CODE                     | Third party postal code                                     |
| Third Party State                        | PARTY_STATE                        | Third party state   |
| Third Party Province                     | PARTY_PROVINCE                     | Third party province  |
| Third Party Country                      | PARTY_COUNTRY                      | Third party country   |
| Third Party County                       | PARTY_COUNTY                       | Third party county  |
| Third Party Site Id                      | PARTY_SITE_ID                      | Third party site internal identifier for journal entry line |
| Third Party Site Number                  | PARTY_SITE_NUMBER                  | Third party site number for journal entry line              |
| Third Party Site Name                    | PARTY_SITE_NAME                    | Third party site name for journal entry line                |
| Third Party Site Tax Registration Number | PARTY_SITE_TAX_REGISTRATION_NUMBER | Third party site taxpayer identifier for journal entry line |
| Third Party Site Address Line 1          | PARTY_SITE_ADDRESS_LINE_1          | Third party site address line 1                             |
| Third Party Site Address Line 2          | PARTY_SITE_ADDRESS_LINE_2          | Third party site address line 2                             |
| Third Party Site Address Line 3          | PARTY_SITE_ADDRESS_LINE_3          | Third party site address line 3                             |
| Third Party Site Address Line 4          | PARTY_SITE_ADDRESS_LINE_4          | Third party site address line 4                             |
| Third Party Site City                    | PARTY_SITE_CITY                    | Third party site city                                       |
| Third Party Site Postal Code             | PARTY_SITE_ZIP_CODE                | Third party site postal code                                |

| Field Name                | Data Element (XML Tag) | Description   |
|---------------------------|------------------------|---|
| Third Party Site State    | PARTY_SITE_STATE       | Third party site state  |
| Third Party Site Province | PARTY_SITE_PROVINCE    | Third party site province                                       |
| Third Party Site Country  | PARTY_SITE_COUNTRY     | Third party site country  |
| Third Party Site County   | PARTY_SITE_COUNTY      | Third party site county   |
| Application Id            | APPLICATION_ID         | Internal identifier of the application owning the journal entry |
| Application Name          | APPLICATION_NAME       | Application name owning the journal entry                       |
| Journal Source Name       | JE_SOURCE_NAME         | Journal entry source  |
| Period Year               | PERIOD_YEAR            | Journal entry fiscal year                                       |
| Period Number             | PERIOD_NUMBER          | Position of the period in the fiscal year                       |
| Period Name               | PERIOD_NAME            | Journal entry accounting period name                            |
| Period Start Date         | PERIOD_START_DATE      | First date of accounting period                                 |
| Period End Date           | PERIOD_END_DATE        | Last date of accounting period                                  |
| Beginning Balance Dr      | BEGIN_BALANCE_DR       | Beginning balance, debit  |
| Beginning Balance Cr      | BEGIN_BALANCE_CR       | Beginning balance, credit                                       |
| Period Activity Dr        | PERIOD_NET_DR          | Period activity, debit  |
| Period Activity Cr        | PERIOD_NET_CR          | Period activity, credit   |

| Field Name                    | Data Element (XML Tag)                                   | Description   |
|-------------------------------|--|---|
| Beginning Draft Balance Dr    | BEGIN_DRAFT_BALANCE_DR                                   | Beginning draft balance, debit  |
| Beginning Draft Balance Cr    | BEGIN_DRAFT_BALANCE_CR                                   | Beginning draft balance, credit   |
| Period Draft Activity Dr      | PERIOD_DRAFT_NET_DR                                      | Period draft activity, debit  |
| Period Draft Activity Cr      | PERIOD_DRAFT_NET_CR                                      | Period draft activity, credit   |
| Code Combination Id           | CODE_COMBINATION_ID                                      | Accounting code combination internal identifier for the journal entry line              |
| Accounting Code Combination   | ACCOUNTING_CODE_COMBINATION (concatenated values)        | Concatenated values of the accounting code combination for the journal entry line       |
| Code Combination Description  | CODE_COMBINATION_DESCRIPTION (concatenated descriptions) | Concatenated descriptions of the accounting code combination for the journal entry line |
| Balancing Segment             | BALANCING_SEGMENT  | Journal entry line balancing segment  |
| Balancing Segment Description | BALANCING_SEGMENT_DESCRIPTION                            | Journal entry line balancing segment description  |
| Natural Account               | NATURAL_ACCOUNT  | Journal entry line natural account segment values                                       |
| Natural Account Description   | NATURAL_ACCOUNT_DESCRIPTION                              | Journal entry line natural account segment values                                       |
| Cost Center                   | COST_CENTER  | Journal entry line cost center segment values   |
| Cost Center Description       | COST_CENTER_DESCRIPTION                                  | Journal entry line cost center segment description                                      |

| Field Name  | Data Element (XML Tag)           | Description  |
|---|----------------------------------|--|
| Management Segment  | MANAGEMENT_SEGMENT               | Journal entry line management segment value                                      |
| Management Segment Description  | MANAGEMENT_SEGMENT_DESCRIPTION   | Journal entry line management segment description                                |
| Inter-company Segment   | INTERCOMPANY_SEGMENT             | Journal entry line intercompany segment value                                    |
| Inter-company Segment Description   | INTERCOMPANY_SEGMENT_DESCRIPTION | Journal entry line intercompany segment description                              |
| Segment 1 (to 30)   | SEGMENT1 (to 30)                 | Journal entry lines individual Accounting Flexfield segment values               |
| Begin Running Total Dr<br>(Populated only if user provides a GL Date range) | BEGIN_RUNNING_TOTAL_DR           | Debit running total from the start date of the first period to the GL Date From  |
| Begin Running Total Cr<br>(Populated only if user provides a GL Date range) | BEGIN_RUNNING_TOTAL_CR           | Credit running total from the start date of the first period to the GL Date From |
| End Running Total Dr<br>(Populated only if user provides a GL Date range)   | END_RUNNING_TOTAL_DR             | Debit running total from the GL Date To to the end date of the last period       |
| End Running Total Cr<br>(Populated only if user provides a GL Date range)   | END_RUNNING_TOTAL_CR             | Credit running total from the GL Date To to the end date of the last period      |

**Third Party Balances Report Journal Entry Header and Line Information**

| Field Name                  | Data Element (XML Tag)      | Source   |
|-----------------------------|-----------------------------|--|
| GL Date                     | GL_DATE                     | Journal entry header accounting date                                     |
| Transaction Created by      | CREATED_BY                  | User who created transaction   |
| Creation Date               | CREATION_DATE               | Journal entry creation date  |
| Last Update Date            | LAST_UPDATE_DATE            | Last date journal entry updated  |
| GL Transfer Date            | GL_TRANSFER_DATE            | Date journal entry transferred to General Ledger                         |
| Reference Date              | REFERENCE_DATE              | Journal entry legal reporting date                                       |
| Completed Date              | COMPLETED_DATE              | Journal entry completion date  |
| Transaction Number          | TRANSACTION_NUMBER          | Subledger transaction number   |
| Transaction Date            | TRANSACTION_DATE            | Subledger transaction date   |
| Accounting Sequence Name    | ACCOUNTING_SEQUENCE_NAME    | Sequence name used to number the journal entry at creation or posting    |
| Accounting Sequence Version | ACCOUNTING_SEQUENCE_VERSION | Sequence version used to number the journal entry at creation or posting |
| Accounting Sequence Number  | ACCOUNTING_SEQUENCE_NUMBER  | Sequence number assigned to the journal entry at creation or posting     |
| Reporting Sequence Name     | REPORTING_SEQUENCE_NAME     | Sequence name used to number the journal entry at period closing         |

| Field Name                 | Data Element (XML Tag)     | Source  |
|----------------------------|----------------------------|---|
| Reporting Sequence Version | REPORTING_SEQUENCE_VERSION | Sequence version used to number the journal entry at period closing           |
| Reporting Sequence Number  | REPORTING_SEQUENCE_NUMBER  | Sequence number assigned to the journal entry at period closing               |
| Document Sequence Name     | DOCUMENT_SEQUENCE_NAME     | Sequence name used to number the subledger transaction                        |
| Document Sequence Number   | DOCUMENT_SEQUENCE_NUMBER   | Sequence number assigned to the subledger transaction                         |
| Header Id                  | HEADER_ID                  | Journal entry internal identifier   |
| Header Description         | HEADER_DESCRIPTION         | Journal entry description   |
| Fund Status Value          | FUND_STATUS                | Fund reservation status for the journal entry                                 |
| Journal Category Name      | JE_CATEGORY_NAME           | Journal entry category  |
| Event Id                   | EVENT_ID                   | Internal identifier of the accounting event that originated the journal entry |
| Event Date                 | EVENT_DATE                 | Date of the accounting event that originated the journal entry                |
| Event Number               | EVENT_NUMBER               | Number of the accounting event that originated the journal entry              |
| Event Class Code           | EVENT_CLASS_CODE           | Event class code of the accounting event that originated the journal entry    |

| Field Name                        | Data Element (XML Tag)               | Source   |
|-----------------------------------|--------------------------------------|--|
| Event Class Name                  | EVENT_CLASS_NAME                     | Event class name of the accounting event that originated the journal entry |
| Event Type Code                   | EVENT_TYPE_CODE                      | Event type code of the accounting event that originated the journal entry  |
| Event Type Name                   | EVENT_TYPE_NAME                      | Event type name of the accounting event that originated the journal entry  |
| User Transaction Identifier Name  | USER_TRX_IDENTIFIER_NA ME_1 (to 10)  | Additional transaction information for identifying the transaction         |
| User Transaction Identifier Value | USER_TRX_IDENTIFIER_VA LUE_1 (to 10) | Additional transaction information for identifying the transaction         |
| Journal Entry Status              | JOURNAL_ENTRY_STATUS                 | Subledger journal entry status: Draft, Final, and Enter                    |
| Transfer to GL Status             | TRANSFER_TO_GL_STATUS                | Indicates whether journal entry transferred to General Ledger              |
| GL Batch Status                   | GL_BATCH_STATUS                      | Status of batch that includes the journal entry: Unposted or Posted        |
| Posted Date                       | POSTED_DATE                          | Date journal entry posted to General Ledger                                |
| Line Number                       | LINE_NUMBER                          | Sequence number assigned to the journal in General Ledger at creation      |
| Accounting Class Code             | ACCOUNTING_CLASS_CO DE               | Accounting class code assigned to the subledger journal entry line         |

| Field Name                 | Data Element (XML Tag)        | Source  |
|----------------------------|-------------------------------|---|
| Accounting Class           | ACCOUNTING_CLASS_NA<br>ME     | Accounting class assigned to the subledger journal entry line |
| Line Description           | LINE_DESCRIPTION              | Journal entry line description                                |
| Entered Currency           | ENTERED_CURRENCY              | Journal entry currency  |
| Conversion Rate            | CONVERSION_RATE               | Transaction conversion rate                                   |
| Conversion Rate Date       | CONVERSION<br>_RATE_DATE      | Transaction conversion rate date                              |
| Conversion Rate Type Code  | CONVERSION<br>_RATE_TYPE_CODE | Transaction conversion type code of rate                      |
| Conversion Rate Type       | CONVERSION_RATE_TYPE          | Transaction conversion rate type                              |
| Entered Debit              | ENTERED_DR                    | Entered debit for journal entry line                          |
| Entered Credit             | ENTERED_CR                    | Entered credit for journal entry line                         |
| Unrounded Accounted Debit  | UNROUNDED_ACCOUNTE<br>D_DR    | Accounted debit before rounding                               |
| Unrounded Accounted Credit | UNROUNDED_ACCOUNTE<br>D_CR    | Accounted credit before rounding                              |
| Accounted Debit            | ACCOUNTED_DR                  | Journal entry line accounted debit                            |
| Accounted Credit           | ACCOUNTED_CR                  | Journal entry line accounted credit                           |
| Statistical Amount         | STATISTICAL_AMOUNT            | Journal entry line statistical amount                         |
| Reconciliation Reference   | RECONCILIATION_REFERE<br>NCE  | Reference used for reconciliation purposes                    |

| Field Name          | Data Element (XML Tag) | Source  |
|---------------------|------------------------|---|
| Attribute Category  | ATTRIBUTE_CATEGORY     | Context column for the journal entry lines descriptive flexfield    |
| Attribute 1 (to 10) | ATTRIBUTE1 (to 10)     | Attribute columns for the journal entry lines descriptive flexfield |

## Multiperiod Accounting Reports

This section describes the following multiperiod Accounting reports:

- Multiperiod Accounting and Accrual Reversal Report: Created when user runs the Create Accounting program and displays accrual, recognition, and accrual reversal journal entries
- Multiperiod Accounting Report (summary or detail): Created when user runs the Complete Multiperiod Accounting program and displays recognition and accrual reversal entries processed by the program

**Note:** For Financials Accounting Hub users who are utilizing a custom view, you must create the custom view under the Applications schema for the Multiperiod Accounting and Accrual Reversal Report to work.

## Multiperiod Accounting and Accrual Reversal Report Data Elements

For information on submission parameters, see Create Accounting Program, page 3-1.

**Note:** Not all data elements have a corresponding field on the seeded multiperiod and accrual reversal report templates.

### *Multiperiod Accounting and Accrual Reversal Report Level Data Elements*

| Field Name         | Data Element (XML Tag) | Description   |
|--------------------|------------------------|---|
| Source Application | SOURCE_APPLICATION     | Source Application value from Create Accounting request parameter |

| Field Name                 | Data Element (XML Tag)     | Description   |
|----------------------------|----------------------------|---|
| Application                | APPLICATION                | Subledger Application value from Create Accounting request parameter  |
| End Date                   | END_DATE                   | End Date value from Create Accounting request parameter; processes only those events with event dates on or before the end date                           |
| Process Category           | PROCESS_CATEGORY           | Process Category value from Create Accounting request parameter; restricts the events for accounting to a particular process category                     |
| Accounting Report Level    | ACCOUNTING_REPORT_LEVEL    | Accounting Report Level value from Create Accounting request parameter  |
| Create Accounting          | CREATE_ACCOUNTING          | Create Accounting value from Create Accounting request parameter; determines whether to create subledger journal entries                                  |
| Errors Only                | ERRORS_ONLY                | Errors Only value from Create Accounting request parameter; limits the creation of accounting to those events which previously failed accounting          |
| Transfer to General Ledger | TRANSFER_TO_GENERAL_LEDGER | Transfer to General Ledger value from Create Accounting request parameter; determines whether to transfer the subledger journal entries to General Ledger |

| Field Name                | Data Element (XML Tag)     | Description  |
|---------------------------|----------------------------|--|
| Post in General Ledger    | POST_IN_GENERAL_LEDGE<br>R | Post in General Ledger value from Create Accounting request parameter; determines whether to post subledger journal entries in General Ledger                              |
| General Ledger Batch Name | GENERAL_LEDGER_BATCH_NAME  | General Ledger Batch Name value from Create Accounting request parameter; user-entered batch name that appears on the transferred General Ledger subledger journal entries |

***Multiperiod Accounting and Accrual Reversal Report Journal Entry Summary Data Elements***

| Field Name          | Data Element (XML Tag) | Description   |
|---------------------|------------------------|---|
| Event Class         | EVENT_CLASS            | Accounting event class name of the subledger journal entries used to group journal entries  |
| Number of Documents | NUMBER_OF_DOCUMENTS    | Number of documents processed for the event class   |
| Ledger              | LEDGER                 | Ledger name of processed journal entries  |
| Type                | JOURNAL_ENTRY_TYPE     | Journal entry type: <ul style="list-style-type: none"> <li>• Multiperiod Accrual</li> <li>• Recognition</li> <li>• Accrual</li> <li>• Accrual Reversal</li> </ul> |

| Field Name | Data Element (XML Tag)    | Description                                       |
|------------|---------------------------|---|
| Processed  | JOURNAL_ENTRIES_PROCESSED | Number of journal entries processed for each type |
| In Error   | JOURNAL_ENTRIES_IN_ERROR  | Number of journal entries with errors             |

***Multiperiod Accounting and Accrual Reversal Report Transfer to General Ledger Data Elements***

| Field Name | Data Element (XML Tag)      | Description   |
|------------|-----------------------------|---|
| Ledger     | LEDGER                      | Ledger for which entries are transferred  |
| Type       | JOURNAL_ENTRY_TYPE          | Journal entry type: <ul style="list-style-type: none"> <li>• Multiperiod Accrual</li> <li>• Recognition</li> <li>• Accrual</li> <li>• Accrual Reversal</li> </ul> |
| Count      | JOURNAL_ENTRIES_TRANSFERRED | Number of journal entries transferred to General Ledger   |

***Multiperiod Accounting and Accrual Reversal Report General Error Data Elements***

| Data Source Element | Template Field  | Description                                |
|---------------------|-----------------|--|
| Error Number        | G_ERROR_NUMBER  | Error code of the accounting program error |
| Error Message       | G_ERROR_MESSAGE | Error description                          |

***Multiperiod Accounting and Accrual Reversal Report Event Level Data Elements***

| Field Name                      | Data Element (XML Tag)         | Description   |
|---------------------------------|--------------------------------|---|
| Event Class                     | EVENT_CLASS                    | Accounting event class name of the subledger journal entries  |
| Event Type                      | EVENT_TYPE                     | Accounting event type of the subledger journal entries  |
| Event Number                    | EVENT_NUMBER                   | User-oriented number to identify the events of a document   |
| Event Date                      | EVENT_DATE                     | Event date of the event   |
| <User Trx Identifier Prompt XX> | USER_TRX_IDENTIFIER_PR OMPT_XX | Prompt values based upon setup in the Accounting Methods Builder (AMB) of Event Class Attribute User Transaction Identifiers. All identifiers set up are displayed. |
| <User Trx Identifier Value XX>  | USER_TRX_IDENTIFIER_VA LUE_XX  | User transaction identifier XX value  |

***Multiperiod Accounting and Accrual Reversal Report Journal Header Data Elements***

| Field Name      | Data Element (XML Tag) | Description                               |
|-----------------|------------------------|---|
| Ledger          | LEDGER                 | Accrual journal entry ledger name         |
| GL Date         | GL_DATE                | Accrual journal entry General Ledger date |
| Ledger Currency | LEDGER_CURRENCY        | Accrual journal entry ledger currency     |

| Field Name                                | Data Element (XML Tag)            | Description   |
|---|-----------------------------------|---|
| Application Accounting Definition         | APPLICATION_ACCOUNTING_DEFINITION | Application accounting definition name used to create accrual journal entry   |
| Application Accounting Definition Version | AAD_VERSION                       | Application accounting definition version of accrual journal entry  |
| Journal Entry Description                 | JOURNAL_ENTRY_DESCRIPTION         | Accrual journal entry description   |
| Type                                      | JOURNAL_ENTRY_TYPE                | Journal entry type: <ul style="list-style-type: none"> <li>• Multiperiod Accrual</li> <li>• Recognition</li> <li>• Accrual</li> <li>• Accrual Reversal</li> </ul> |
| Journal Entry Status                      | JOURNAL_ENTRY_STATUS              | Accrual journal entry status  |
| Accounting Sequence Name                  | ACCOUNTING_SEQUENCE_NAME          | Sequence name used to number the journal entry at completion time   |
| Accounting Sequence Version               | ACCOUNTING_SEQUENCE_VERSION       | Sequence version used to number the journal entry at completion time  |
| Accounting Sequence Number                | ACCOUNTING_SEQUENCE_NUMBER        | Sequential number assigned to the journal entry at completion time  |

***Multiperiod Accounting and Accrual Reversal Report Journal Entry Line Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>                      |
|-------------------|-------------------------------|---|
| Line Number       | LINE_NUMBER                   | Accrual journal entry line number       |
| Account           | ACCOUNT                       | Accrual journal entry line account      |
| Currency          | CURRENCY                      | Transaction line entered currency code  |
| Entered Debit     | ENTERED_DEBIT                 | Transaction line currency debit amount  |
| Entered Credit    | ENTERED_CREDIT                | Transaction line currency credit amount |
| Accounted Debit   | ACCOUNTED_DEBIT               | Ledger currency debit amount            |
| Accounted Credit  | ACCOUNTED_CREDIT              | Ledger currency credit amount           |

***Multiperiod Accounting and Accrual Accounting Report Error Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>            |
|-------------------|-------------------------------|-------------------------------|
| Line Number       | LINE_NUMBER                   | Error line number             |
| Error Number      | ERROR_NUMBER                  | Accounting program error code |
| Error Message     | ERROR_MESSAGE                 | Error description             |

## Multiperiod Accounting Report (Detail) Data Elements

### *Multiperiod Accounting Report (Detail) Report Level Data Elements*

| Field Name                 | Data Element (XML Tag)     | Description   |
|----------------------------|----------------------------|---|
| Application                | APPLICATION                | Subledger Application value from Complete Multiperiod Accounting request parameter      |
| End Date                   | END_DATE                   | End Date value from Complete Multiperiod Accounting request parameter                   |
| Process Category           | PROCESS_CATEGORY           | Process Category value from Complete Multiperiod Accounting request parameter           |
| Errors Only                | ERRORS_ONLY                | Errors Only value from Complete Multiperiod Accounting request parameter                |
| Transfer to General Ledger | TRANSFER_TO_GENERAL_LEDGER | Transfer to General Ledger value from Complete Multiperiod Accounting request parameter |
| Post in General Ledger     | POST_IN_GENERAL_LEDGER     | Post in General Ledger value from Complete Multiperiod Accounting request parameter     |
| General Ledger Batch Name  | GENERAL_LEDGER_BATCH_NAME  | General Ledger Batch Name value from Complete Multiperiod Accounting request parameter  |

***Multiperiod Accounting Report (Detail) Journal Entry Summary Data Elements***

| <b>Field Name</b>   | <b>Data Element (XML Tag)</b> | <b>Description</b>  |
|---------------------|-------------------------------|---|
| Event Class         | EVENT_CLASS                   | Accounting event class name of the subledger journal entries used to group journal entries                      |
| Number of Documents | NUMBER_OF_DOCUMENTS           | Number of documents processed for the event class   |
| Ledger              | LEDGER                        | Processed journal entries ledger name   |
| Type                | JOURNAL_ENTRY_TYPE            | Journal entry type: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Processed           | JOURNAL_ENTRIES_PROCESSED     | Number of journal entries processed for each type   |
| In Error            | JOURNAL_ENTRIES_IN_ERROR      | Number of journal entries with errors   |

***Multiperiod Accounting Report (Detail) Transfer to General Ledger Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>   |
|-------------------|-------------------------------|--|
| Ledger            | LEDGER                        | Ledger for which entries are transferred   |
| Type              | JOURNAL_ENTRY_TYPE            | Type of journal entry: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Count             | JOURNAL_ENTRIES_TRANSFERRED   | Number of journal entries transferred to General Ledger  |

***Multiperiod Accounting Report (Detail) General Error Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b> |
|-------------------|-------------------------------|--------------------|
| Error Number      | G_ERROR_NUMBER                | Program error code |
| Error Message     | G_ERROR_MESSAGE               | Error description  |

***Multiperiod Accounting Report (Detail) Event Level Data Elements***

| <b>Field Name</b>               | <b>Data Element (XML Tag)</b> | <b>Description</b>  |
|---------------------------------|-------------------------------|---|
| Event Class                     | EVENT_CLASS                   | Accounting event class name of the subledger journal entries  |
| Event Type                      | EVENT_TYPE                    | Accounting event type of the subledger journal entries  |
| Event Number                    | EVENT_NUMBER                  | Number to identify the events of a document   |
| Event Date                      | EVENT_DATE                    | Date of the event   |
| <User Trx Identifier Prompt XX> | USER_TRX_IDENTIFIER_PROMPT_XX | Prompt values based upon setup in the Accounting Methods Builder (AMB) of Event Class Attribute User Transaction Identifiers. All identifiers set up are displayed. |
| <User Trx Identifier Value XX>  | USER_TRX_IDENTIFIER_VALUE_XX  | Value for the user transaction identifier XX  |

***Multiperiod Accounting Report (Detail) Journal Entry Header Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>        |
|-------------------|-------------------------------|---------------------------|
| Ledger            | LEDGER                        | Journal entry ledger name |

| Field Name                                | Data Element (XML Tag)            | Description   |
|---|-----------------------------------|---|
| GL Date                                   | GL_DATE                           | Journal entry General Ledger date   |
| Ledger Currency                           | LEDGER_CURRENCY                   | Journal entry ledger currency   |
| Application Accounting Definition         | APPLICATION_ACCOUNTING_DEFINITION | Application accounting definition name used to create journal entry   |
| Application Accounting Definition Version | AAD_VERSION                       | Application accounting definition version   |
| Description                               | DESCRIPTION                       | Journal entry description   |
| Type                                      | JOURNAL_ENTRY_TYPE                | Journal entry type: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Journal Entry Status                      | JOURNAL_ENTRY_STATUS              | Journal entry status  |
| Accounting Sequence Name                  | ACCOUNTING_SEQUENCE_NAME          | Sequence name used to number the journal entry at completion time   |
| Accounting Sequence Version               | ACCOUNTING_SEQUENCE_VERSION       | Sequence version used to number the journal entry at completion time  |
| Accounting Sequence Number                | ACCOUNTING_SEQUENCE_NUMBER        | Sequential number assigned to the journal entry at completion time  |

#### ***Multiperiod Accounting Report (Detail) Journal Entry Line Data Elements***

| Field Name  | Data Element (XML Tag) | Description               |
|-------------|------------------------|---------------------------|
| Line Number | LINE_NUMBER            | Journal entry line number |

| Field Name       | Data Element (XML Tag) | Description                                      |
|------------------|------------------------|--|
| Account          | ACCOUNT                | Journal entry line account                       |
| Currency         | CURRENCY               | Journal entry line entered currency code         |
| Entered Debit    | ENTERED_DEBIT          | Journal entry line entered currency debit amount |
| Entered Credit   | ENTERED_CREDIT         | Journal line entered currency credit amount      |
| Accounted Debit  | ACCOUNTED_DEBIT        | Ledger currency debit amount                     |
| Accounted Credit | ACCOUNTED_CREDIT       | Ledger currency credit amount                    |

#### ***Multiperiod Accounting Report (Detail) Error Data Elements***

| Field Name    | Data Element (XML Tag) | Description                   |
|---------------|------------------------|-------------------------------|
| Line Number   | LINE_NUMBER            | Error line number             |
| Error Number  | ERROR_NUMBER           | Accounting program error code |
| Error Message | ERROR_MESSAGE          | Error description             |

## Multiperiod Accounting Report (Summary) Data Elements

### *Multiperiod Accounting Report (Summary) Report Level Data Elements*

| Field Name                 | Data Element (XML Tag)     | Description   |
|----------------------------|----------------------------|---|
| Application                | APPLICATION                | Subledger Application value from Complete Multiperiod Accounting request parameter      |
| End Date                   | END_DATE                   | End Date value from Complete Multiperiod Accounting request parameter                   |
| Process Category           | PROCESS_CATEGORY           | Process Category value from Complete Multiperiod Accounting request parameter           |
| Errors Only                | ERRORS_ONLY                | Errors Only value from Complete Multiperiod Accounting request parameter                |
| Transfer to General Ledger | TRANSFER_TO_GENERAL_LEDGER | Transfer to General Ledger value from Complete Multiperiod Accounting request parameter |
| Post in General Ledger     | POST_IN_GENERAL_LEDGER     | Post in General Ledger value from Complete Multiperiod Accounting request parameter     |
| General Ledger Batch Name  | GENERAL_LEDGER_BATCH_NAME  | General Ledger Batch Name value from Complete Multiperiod Accounting request parameter  |

***Multiperiod Accounting Report (Summary) Journal Entry Summary Data Elements***

| <b>Field Name</b>   | <b>Data Element (XML Tag)</b> | <b>Description</b>  |
|---------------------|-------------------------------|---|
| Event Class         | EVENT_CLASS                   | Accounting event class name of the subledger journal entries used to group journal entries                      |
| Number of Documents | NUMBER_OF_DOCUMENTS           | Number of documents processed for the event class   |
| Ledger              | LEDGER                        | Processed journal entries ledger name   |
| Type                | JOURNAL_ENTRY_TYPE            | Journal entry type: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Processed           | JOURNAL_ENTRIES_PROCESSED     | Number of journal entries processed for each type   |
| In Error            | JOURNAL_ENTRIES_IN_ERROR      | Number of journal entries with errors   |

***Multiperiod Accounting Report (Summary) Transfer to General Ledger Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>  |
|-------------------|-------------------------------|---|
| Ledger            | LEDGER                        | Ledger for which entries are transferred  |
| Type              | JOURNAL_ENTRY_TYPE            | Journal entry type: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Count             | JOURNAL_ENTRIES_TRANSFERRED   | Number of journal entries transferred to General Ledger   |

***Multiperiod Accounting Report (Summary) General Error Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>            |
|-------------------|-------------------------------|-------------------------------|
| Error Number      | G_ERROR_NUMBER                | Accounting program error code |
| Error Message     | G_ERROR_MESSAGE               | Error description             |

***Multiperiod Accounting Report (Summary) Journal Entry Header Data Elements***

| <b>Field Name</b>                         | <b>Data Element (XML Tag)</b>     | <b>Description</b>  |
|---|-----------------------------------|---|
| Ledger                                    | LEDGER                            | Journal entry ledger  |
| GL Date                                   | GL_DATE                           | Journal entry General Ledger date   |
| Ledger Currency                           | LEDGER_CURRENCY                   | Journal entry ledger currency   |
| Application Accounting Definition         | APPLICATION_ACCOUNTING_DEFINITION | Application accounting definition name used to create journal entry   |
| Application Accounting Definition Version | AAD_VERSION                       | Application accounting definition version   |
| Description                               | DESCRIPTION                       | Journal entry description   |
| Type                                      | JOURNAL_ENTRY_TYPE                | Journal entry type: <ul style="list-style-type: none"> <li>• Recognition</li> <li>• Accrual Reversal</li> </ul> |
| Journal Entry Status                      | JOURNAL_ENTRY_STATUS              | Journal entry status  |

#### ***Multiperiod Accounting Report (Summary) Journal Entry Line Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>                               |
|-------------------|-------------------------------|--|
| Line Number       | LINE_NUMBER                   | Journal entry line number                        |
| Account           | ACCOUNT                       | Journal entry line account                       |
| Currency          | CURRENCY                      | Journal entry line entered currency code         |
| Entered Debit     | ENTERED_DEBIT                 | Journal entry line entered currency debit amount |
| Entered Credit    | ENTERED_CREDIT                | Journal entry line currency credit amount        |
| Accounted Debit   | ACCOUNTED_DEBIT               | Ledger currency debit amount                     |
| Accounted Credit  | ACCOUNTED_CREDIT              | Ledger currency credit amount                    |

#### ***Multiperiod Accounting Report (Summary) Error Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>            |
|-------------------|-------------------------------|-------------------------------|
| Line Number       | LINE_NUMBER                   | Error line number             |
| Error Number      | ERROR_NUMBER                  | Accounting program error code |
| Error Message     | ERROR_MESSAGE                 | Error description             |

## **Subledger Period Close Exceptions Report**

The Subledger Period Close Exceptions Report lists all accounting events and journal entries that fail period close validation. It is automatically submitted by General Ledger when closing a GL period if there are unprocessed accounting events or untransferred journal entries.

You can also generate the Subledger Period Close Exceptions Report through a concurrent request as follows:

- For the application associated with the responsibility
- For all applications in the General Ledger responsibility

## Subledger Period Close Exceptions Report Parameters

### ***Subledger Period Close Exceptions Report Parameters***

| <b>Parameter</b>   | <b>Required</b>   | <b>Valid Values</b>   |
|--------------------|---|---|
| Ledger /Ledger Set | Yes   | All ledgers and ledger sets included in the access set identified by the profile options GL: Data Access Set and SLA: Additional Access Set |
| Period From        | Yes   | All periods from the accounting calendar associated with the selected ledger or ledger set  |
| Period To          | Yes   | All periods from the accounting calendar equal to or coming after the From period up to the end of the fiscal year                          |
| Journal Source     | No  | All General Ledger journal sources registered with Subledger Accounting   |
| Event Class        | No; enabled only when a concurrent request is submitted for a specific journal source | All event classes enabled for the application associated with the journal source  |
| Journal Category   | No; enabled only when a concurrent request is submitted for a journal source          | All journal categories used by Subledger Accounting   |

## Subledger Period Close Exceptions Report Data Elements

**Note:** Not all data elements have a corresponding field on the seeded subledger period close exceptions report templates.

### ***Subledger Period Close Exceptions Report Data Elements***

| Field Name             | Data Element (XML Tag) | Description   |
|------------------------|------------------------|---|
| Event Date             | EVENT_DATE             | Event date; if journal entry exists, journal entry GL date  |
| Transaction Created by | CREATED_BY             | User who submitted the Subledger Accounting program   |
| Creation Date          | CREATION_DATE          | Event creation date; if journal entry exists, journal entry creation date   |
| Last Update Date       | LAST_UPDATE_DATE       | Last date event updated; if journal entry exists, last date journal entry updated   |
| User Name              | USER_NAME              | In case of events, name of user who created the event; in case of journal entries, name of user who created the journal entries |
| Transaction Number     | TRANSACTION_NUMBER     | Subledger transaction number  |
| Transaction Date       | TRANSACTION_DATE       | Subledger transaction date  |
| Application Id         | APPLICATION_ID         | Internal identifier of the application owning the journal entry or accounting event   |
| Ledger ID              | LEDGER_ID              | Internal identifier of the ledger owning the journal entry or accounting event  |
| Ledger Short Name      | LEDGER_SHORT_NAME      | Short name of the ledger owning the journal entry or accounting event   |

| Field Name            | Data Element (XML Tag) | Description   |
|-----------------------|------------------------|---|
| Ledger Description    | LEDGER_DESCRIPTION     | Description of the ledger owning the subledger journal entry  |
| Ledger Name           | LEDGER_NAME            | Name of the ledger owning the subledger journal entry or accounting event   |
| Ledger Currency       | LEDGER_CURRENCY        | Currency of the ledger owning the subledger journal entry or accounting event   |
| Status                | PRINT_STATUS           | Accounting event status if no journal entries exist   |
|                       |                        | If journal entry exists: if it is accounted in final mode, status of the General Ledger transfer status code; otherwise status of the journal entry |
| On Hold Flag          | ON_HOLD_FLAG           | Indicates whether event is on hold; Y for Yes, N for No   |
| On Hold               | ON_HOLD_FLAG           | Values are Yes or No.   |
| Journal Category Name | JOURNAL_CATEGORY_NAME  | Journal category name associated with the event class of the event  |
|                       | USER_JE_CATEGORY_NAME  | Journal category name associated with the event class of the event in user's language   |
| Journal Source Name   | JOURNAL_SOURCE         | Journal source associated with the application owning the journal entry   |
|                       | USER_JE_SOURCE         | Journal source associated with the application owning the journal entry in user's language  |
| Event Id              | EVENT_ID               | Accounting event internal identifier  |

| Field Name                        | Data Element (XML Tag)              | Description  |
|-----------------------------------|-------------------------------------|--|
| Event Date                        | ACCOUNTING_DATE                     | Event date of the accounting event; if journal entry exists, accounting date of the journal entry  |
| Event Number                      | EVENT_NUMBER                        | Accounting event number  |
| Event Class Code                  | EVENT_CLASS_CODE                    | Event class code of the accounting event   |
| Event Class Name                  | EVENT_CLASS_NAME                    | Event class name of the accounting event   |
| Event Type Code                   | EVENT_TYPE_CODE                     | Event type code of the accounting event  |
| Event Type Name                   | EVENT_TYPE_NAME                     | Event type name of the accounting event  |
| User Transaction Identifier Name  | USER_TRX_IDENTIFIER_NAME_1 (to 10)  | User transaction identifiers column names attached to the event class of the accounting event  |
| User Transaction Identifier Value | USER_TRX_IDENTIFIER_VALUE_1 (to 10) | User transaction identifiers values attached to the event class of the accounting event  |
| Balance Type                      | BALANCE_TYPE                        | Valid only for journal entries and not for events; balance type of the journal entry. Possible values include Actual, Budget, and Encumbrance. |
| Balance Type Code                 | BALANCE_TYPE_CODE                   | A for Actual, B for Budget, and E for Encumbrance  |

## Subledger Accounting Rules Detail Listing Report

The Subledger Accounting Rules Detail Listing Report lets you view setup details for a subledger accounting method.

The report can be submitted from individual subledger applications or from Oracle General Ledger. In addition, the report can be submitted for custom applications registered within Oracle Financial Accounting Hub.

**Note:** This report is not available by default with the standard request groups. You must attach this report to the required request group in the System Administrator responsibility.

## Subledger Accounting Rules Detail Listing Report Parameters

### *Subledger Accounting Rules Detail Listing Report Parameters*

| Parameter                   | Required  | Valid Values   |
|-----------------------------|---|--|
| Owner Type                  | Yes   | A valid E-Business Suite user.   |
| Subledger Accounting Method | Yes   | The list of values shows the Subledger Accounting Method names that are Enabled. The value set is based on the value selected in the Owner Type. |
| Application                 | Yes   | Application Name attached to the responsibility from where the report is submitted.  |
| Event Class                 | No; enabled only when a concurrent request is submitted for a specific journal source | All event classes enabled for the application associated with the journal source   |
| Event Type                  | No; enabled only when a concurrent request is submitted for a specific journal source | Event type code of the accounting event  |

## Subledger Accounting Rules Detail Listing Report Data Element

**Note:** Not all data elements have a corresponding field on the seeded subledger accounting rules detail listing report templates.

### ***Subledger Accounting Rules Detail Listing Report Data Elements***

| <b>Field Name</b> | <b>Data Element (XML Tag)</b> | <b>Description</b>  |
|-------------------|-------------------------------|---|
| Owner Type        | CREATED_BY                    | User who submitted the Subledger Accounting program                                 |
| Application       | APPLICATION_ID                | Internal identifier of the application owning the journal entry or accounting event |
| Event Class Code  | EVENT_CLASS_CODE              | Event class code of the accounting event  |
| Event Type        | EVENT_TYPE_CODE               | Event type code of the accounting event   |

The report output comprises of header, subheader and appendix. You can now use the new report to easily review the following setup details:

- Subledger Accounting Method
- Journal Line Definition
- Journal Line Type
- Account Derivation Rule
- Journal Header Description
- Journal Line Description

---

## Open Account Balances Listing

### Open Account Balances Listing

The Open Account Balances Listing identifies General Ledger accounts with outstanding balances and displays the subledger transactions that contribute to that balance. Users can create their own layout and publish their reports using Oracle XML Publisher.

See:

- Subledger Accounting Reports Introduction, page 9-1
- Oracle XML Publisher Introduction, *Oracle XML Publisher User's Guide*
- Subledger Accounting Setup Options Description, page 1-1

### Open Account Balances Listing Process Steps

1. Create open account balance definitions in the Create Open Account Balances Listing Definition page.

See: Open Account Balance Listing Definitions, page 10-2

Optionally, to update Open Account Balances Listing processing options, navigate to the Update Open Account Balances Listing Processing Options page by clicking **Processing Options** in the Open Account Balances Listing Definition Search page.

For each ledger used for open account balance reporting, Subledger Accounting defaults the number of processors to 1 and the processing unit size to 5000. Change these defaults to improve the performance of the Open Account Balances Listing Data Manager.

**Note:** A processing unit is the number of transactions processed by

the Open Account Balances Listing Data Manager in one commit cycle.

2. Create subledger journal entries and transfer them to General Ledger.
3. After journal entries are transferred to General Ledger, Subledger Accounting automatically submits the Open Account Balances Listing Data Manager program, which maintains the records in the open account balances tables according to the open account balances listing definitions. Users can also manually submit this program as a concurrent request.

See: Open Account Balances Data Manager, page 10-5
4. Submit the Open Account Balances Listing.

See: Open Account Balances Listing, page 10-5

## Creating Open Account Balances Listing Definitions

Users define the accounts to be tracked in the Open Account Balances Listing in the Create Open Account Balances Listing Definition page. Define the report definitions either by Accounting Flexfield or by Accounting Flexfield segments. Associate each definition with a single ledger, which indicates the chart of accounts. The chart of accounts sets the context for the following information in the definition details region:

- General Ledger accounts
- General Ledger account segments
- General Ledger account segment values

Define as many definitions as necessary. If defining a report definition by segment, indicate a single segment value, a range of segment values, or a combination of individual values and ranges for different Accounting Flexfield segments. If defining a report by Accounting Flexfield, include one or more entire General Ledger accounts.

## Prerequisites

- Define a ledger.

See: Primary Ledger Setup Steps, *Oracle Financials Implementation Guide*
- Create journal entries and transfer to General Ledger using the Create Accounting program.

See: Create Accounting Program, page 3-1

- Identify links between source transactions and applied transactions by mapping appropriate sources to the Applied-to accounting attributes.

**Note:** The Open Account Balances Listing relies on the Applied-to accounting attributes, which also support the business flow feature, to determine how an applied transaction affects the outstanding balance of a source transaction.

**Note:** Do not create Open Account Balances Definition Codes using SQL reserved keywords as same codes are used to create partitions on Subledger Accountning Trial Balances table internally. This results in an error.

## To Create or Update Open Account Balances Listing Definitions

The screenshot shows the Oracle Subledger Accounting Setups interface. The main title is "Open Account Balances Listing Definitions". Below it is a "Simple Search" section with fields for Code, Name, and Ledger, and dropdowns for Defined By (Segment), Balance Side (Credit/Debit), and Enabled (Yes/No). There are "Go" and "Clear" buttons. Below the search is a "Create" section with a table for entering new data. The table has columns: Details, Code, Name, Ledger, Defined By, Balance Side, Enabled, Description, and Actions. A row is shown with a checkbox (unchecked), Code A1, Name A1, Ledger SLA East Palo Alto Corporate, Defined By Segment, Balance Side Credit, Enabled Yes, and Actions with edit, copy, and delete icons. Below this is a section for "Open Account Balances Listing Journal Sources" with a "Journal Source" field containing "SLA SFO". At the bottom is a "Open Account Balances Listing Definition Details" section for "Chart Of Accounts SLA SFO Corporate" with a table for "Segment" and "Account" with "Value From" 1000001 and "Value To" 1234500. At the very bottom are links for "About this Page", "Privacy Statement", and copyright information "Copyright (c) 2008, Oracle. All rights reserved".

When the ledger is defined, the details region displays data according to the value selected in the Defined By field.

**Selected Fields in the Subledger Open Account Balances Listing Definitions Page**

| Field Name          | Description   |
|---------------------|---|
| Code                | Unique internal identifier<br><br><b>Note:</b> For technical reasons, the Open Account Balances Listing code cannot contain spaces.   |
| Defined By          | The display in the Details region is determined by the option selected  |
| Journal Source      | Limits information displayed in the report to transactions belonging to the subledger using this journal source   |
| Balance Side        | Determines how applied transactions affect the balance of an account impacted by the original transaction. When the balance side is credit, all debit entries are subtracted from the credit amount to determine the amount remaining. If the balance side is Debit, all credit entries are subtracted from the debit amount to determine the amount remaining. |
| Enabled             | Indicates whether the report definition is enabled. A disabled report definition does not gather, record, or report open account balance data.  |
| Segment             | Displayed if the Defined By field is Segment. The default is the segment flagged as the natural account segment qualifier.  |
| Value From/Value To | Displayed if the Defined By field is Segment; specifies the range of segment values to be considered for open account balance reporting. To specify a single value, enter the same value in both fields.  |

**Note:** You cannot update the following fields:

- Code
- Ledger
- Defined By
- Journal Source
- Balance Side

## Open Account Balances Data Manager

The Open Account Balances Data Manager maintains reportable information for all enabled open account balance listing definitions. This program is submitted automatically after a successful transfer to General Ledger for the same ledger or manually by running the Open Account Balances Data Manager program. When changes are applied to a Open Account Balances Listing Definition, the Open Account Balances Data Manager program is automatically submitted for the changed definition.

The Open Account Balances Data Manager processes subledger journal entries that are transferred to General Ledger and also final accounted journal entries that are flagged as No Transfer because these entries are already represented in General Ledger.

### To Run the Open Account Balances Data Manager Program

1. In the Submit Request window, select Open Account Balance Data Manager.
2. Enter parameters as described in the table below.

| Parameter         | Description  |
|-------------------|--|
| Ledger            | Provides the context for handling open account balance reportable data |
| Report Definition | Limits processing to the report definition entered                     |

## Open Account Balances Listing

The Open Account Balances Listing displays a General Ledger account, its balance, and a listing of the transactions that contribute to that balance. This report displays the actual accounted balance and does not include encumbrance or budgetary accounts in

General Ledger. Users can view the report in detail or summary mode.

This report displays information that originates from subledger journal entries only. Journal entries created manually in General Ledger or anywhere outside of Subledger Accounting are not included. Only transactions with nonzero outstanding balances are displayed.

Subledger Accounting seeds the following templates:

- Group by Account, Applied Transaction Detail

This template groups the output by General Ledger account. For each account, the report shows the General Ledger balance and the transactions (grouped by third party) that contribute to the balance of that account.

- Group by Account, Summarized

This template groups and summarizes the output by General Ledger account.

- Group by Third Party, Applied Transaction Detail

This template groups output by third party. For each third party, the report displays the General Ledger accounts and transactions with outstanding balances.

**Note:** For the Group by Third Party templates, General Ledger account balances are not displayed because General Ledger does not track individual balances for third parties, only the net balances of an account.

- Group by Third Party, Summarized

This template groups the output by third party with one row summarizing each account. No transactions are displayed. For each third party, one row is displayed for every General Ledger account having an outstanding balance.

## Submission Parameters

### *Open Account Balances Listing Submission Parameters*

| Parameter         | Required | Valid Values  |
|-------------------|----------|---|
| Report Definition | Yes      | All enabled report definitions; displays definition name, ledger, description, and code |

| Parameter                       | Required | Valid Values  |
|---------------------------------|----------|---|
| Journal Source                  | No       | <p>Journal source attached to the application submitting the report. Can be changed only if the SLA: Allow Reports Journal Source Override profile option is enabled.</p> <p>See: SLA: Allow Reports Journal Source Override, page B-8</p> <p>List of values lists all journal sources assigned to the report definition.</p> |
| Start Date                      | Yes      | Defaults to first date of the current fiscal year   |
| As Of Date                      | No       | GL date of journal entries up to which the open account balance listing will include  |
| Third Party Name                | No       | Third parties whose type is either Customer or Supplier. The list of values lists third party name and third party type.  |
| Show Applied Transaction Detail | Yes      | Yes or No; determines whether output displays parent and dependent transaction information. Default is No.  |
| Include Write Off Balances      | Yes      | Yes or No; determines whether to show outstanding balances that are the result of write-off accounting. Default is No.  |
| Account From/To                 | No       | Limits report output by filtering values entered here against reportable values assigned to the report definition   |
| Account Balance                 | Yes      | List of values options are Year to Date and Current Period.   |

**Note:** Use the following custom parameters to pass information to the template.

#### **Open Account Balances Listing Custom Parameters**

| Parameter                       | Required | Valid Values                                    |
|---------------------------------|----------|---|
| Security Parameter              | No       | Hidden parameter for use by uptaking subledgers |
| Valuation Method                | No       | No validation                                   |
| Security Identifier (Number)    | No       | No validation                                   |
| Security Identifier (Character) | No       | No validation                                   |
| Custom Parameter 1              | No       | No validation                                   |
| Custom Parameter 2              | No       | No validation                                   |
| Custom Parameter 3              | No       | No Validation                                   |

#### **Open Account Balances Listing Data Elements**

The table below describes the information that is retrieved by the Open Account Balances Listing. Users can create new XML Publisher templates using any of these data elements.

See: Creating a Template, *Oracle XML Publisher User's Guide*

**Note:** Not all data elements have a corresponding field on the seeded open account balance templates. For the fields not included on the seeded templates, the field name column shows the suggested label for the field.

#### **Open Account Balances Listing Data Elements**

| Template Field Name | Data Element (XML Tag) | Source / Description           |
|---------------------|------------------------|--------------------------------|
| Account             | ACCOUNT                | Accounting Flexfield value     |
| GL Balance          | GL_BALANCE             | General Ledger account balance |

| Template Field Name | Data Element (XML Tag)        | Source / Description   |
|---------------------|-------------------------------|--|
|                     | CODE_COMBINATION_ID           | Account code combination identifier                                      |
|                     | BALANCING_SEGMENT_VALUE       | Balancing segment value  |
|                     | BALANCING_SEGMENT_DESC        | Balancing segment value description                                      |
|                     | NATURAL_ACCOUNT_SEGMENT_VALUE | Natural account segment value  |
|                     | NATURAL_ACCOUNT_SEGMENT_DESC  | Natural account segment value description                                |
|                     | COST_CENTER_SEGMENT_VALUE     | Cost center segment value  |
|                     | COST_CENTER_SEGMENT_DESCR     | Cost center segment value description                                    |
|                     | MANAGEMENT_SEGMENT_VALUE      | Management segment value   |
|                     | MANAGEMENT_SEGMENT_DESC       | Management segment value description                                     |
|                     | INTERCOMPANY_SEGMENT_VALUE    | Intercompany segment value   |
|                     | INTERCOMPANY_SEGMENT_DESC     | Intercompany segment value description                                   |
|                     | LEDGER_ID                     | Internal identifier of the ledger corresponding to the report definition |
| Ledger              | LEDGER_NAME                   | Ledger name  |
|                     | LEDGER_SHORT_NAME             | Ledger short name  |

| Template Field Name | Data Element (XML Tag)         | Source / Description  |
|---------------------|--------------------------------|---|
| Curr                | LEDGER_CURRENCY_CODE           | Ledger currency code  |
| Third Party         | THIRD_PARTY_NAME               | Third party name referenced on the source transaction                       |
|                     | THIRD_PARTY_NUMBER             | Number assigned to the third party referenced on the source transaction     |
|                     | THIRD_PARTY_TYPE               | Third party type for the source transaction                                 |
|                     | THIRD_PARTY_TYPE_CODE          | Third party type code for the source transaction                            |
|                     | THIRD_PARTY_SITE_NAME          | Third party site name referenced on the source transaction                  |
|                     | SOURCE_ENTITY_ID               | Source entity internal identifier   |
|                     | SOURCE_TRX_APPLICATION_ID      | Internal identifier of the application that owns the source transaction     |
|                     | SOURCE_TRX_APPLICATION_NAME    | Name of the application that owns the source transaction                    |
| Transaction Type    | SOURCE_TRX_TYPE                | Source transaction type   |
| Transaction Number  | SOURCE_TRX_NUMBER              | Source transaction number   |
| Date                | SOURCE_TRX_GL_DATE             | Source transaction GL date  |
|                     | SOURCE_TRX_CURR                | Source transaction currency   |
|                     | SRC_ENTERED_UNROUNDED_ORIG_AMT | Unrounded original amount of the source transaction in the entered currency |

| Template Field Name | Data Element (XML Tag)        | Source / Description   |
|---------------------|-------------------------------|--|
|                     | SRC_ENTERED_UNROUNDED_REM_AMT | Unrounded remaining amount of the source transaction in the entered currency |
|                     | SRC_ENTERED_ROUNDED_ORIG_AMT  | Rounded original amount of the source transaction in the entered currency    |
|                     | SRC_ENTERED_ROUNDED_Rem_AMT   | Rounded remaining amount of the source transaction in the entered currency   |
|                     | SRC_ACCTD_UNROUNDED_ORIG_AMT  | Unrounded original amount of the source transaction in the ledger currency   |
|                     | SRC_ACCTD_UNROUNDED_Rem_AMT   | Unrounded remaining amount of the source transaction in the ledger currency  |
| Original Amount     | SRC_ACCTD_ROUNDED_ORIG_AMT    | Rounded original amount of the source transaction in the ledger currency     |
| Remaining Amount    | SRC_ACCTD_ROUNDED_Rem_AMT     | Rounded remaining amount of the source transaction in the ledger currency    |
|                     | APPLIED_TO_APPLICATION_ID     | Internal identifier of the application that owns the applied transaction     |
|                     | USER_TRX_IDENTIFIER_NAME_1    | Source transaction identifier name   |
|                     | USER_TRX_IDENTIFIER_NAME_2    | Source transaction identifier name   |
|                     | USER_TRX_IDENTIFIER_NAME_3    | Source transaction identifier name   |

| Template Field Name | Data Element (XML Tag)          | Source / Description                   |
|---------------------|---------------------------------|--|
|                     | USER_TRX_IDENTIFIER_NA<br>ME_4  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_5  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_6  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_7  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_8  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_9  | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_NA<br>ME_10 | Source transaction identifier<br>name  |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_1 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_2 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_3 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_4 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_5 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_6 | Source transaction identifier<br>value |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_7 | Source transaction identifier<br>value |

| Template Field Name | Data Element (XML Tag)            | Source / Description  |
|---------------------|-----------------------------------|---|
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_8   | Source transaction identifier value                                 |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_9   | Source transaction identifier value                                 |
|                     | USER_TRX_IDENTIFIER_VA<br>LUE_10  | Source transaction identifier value                                 |
|                     | APPLIED_THIRD_PARTY_N<br>AME      | Third party name on the applied transaction                         |
|                     | APPLIED_THIRD_PARTY_N<br>UMBER    | Third party number on the applied transaction                       |
|                     | APPLIED_THIRD_PARTY_T<br>YPE      | Third party type on the applied transaction                         |
|                     | APPLIED_THIRD_PARTY_T<br>YPE_CODE | Third party type code on the applied transaction                    |
|                     | APPLIED_THIRD_PARTY_SI<br>TE_NAME | Third party site name on the applied transaction                    |
|                     | APPLIED_TO_ENTITY_ID              | Entity identifier to which the transaction is applied               |
|                     | APPLIED_TRX_APPLICATIO<br>N_ID    | Application internal identifier to which the transaction is applied |
| Application         | APPLIED_TRX_APPLICATIO<br>N_NAME  | Name of the application that owns the applied transaction           |
| Type                | APPLIED_TRX_TYPE                  | Applied transaction type  |
| Number              | APPLIED_TRX_NUMBER                | Applied transaction number  |
| Date                | APPLIED_TRX_GL_DATE               | Applied transaction GL date   |
|                     | APPLIED_TRX_CURR                  | Applied transaction currency  |

| Template Field Name | Data Element (XML Tag)             | Source / Description   |
|---------------------|------------------------------------|--|
|                     | APPLIED_ENTERED_UNROUNDED_AMT      | Unrounded amount applied to the source transaction in the entered currency |
|                     | APPLIED_ENTERED_ROUNDED_AMT        | Rounded amount applied to the source transaction in the entered currency   |
|                     | APPLIED_ACCTD_UNROUNDED_AMT        | Unrounded amount applied to the source transaction in the ledger currency  |
| Applied Amount      | APPLIED_ACCTD_ROUNDED_AMT          | Rounded amount applied to the source transaction in the ledger currency    |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_1 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_2 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_3 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_4 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_5 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_6 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_7 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_8 | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDENTIFIER_NAME_9 | Applied transaction identifier name  |

| Template Field Name | Data Element (XML Tag)                   | Source / Description                 |
|---------------------|--|--------------------------------------|
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_NAME_10  | Applied transaction identifier name  |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_1  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_2  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_3  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_4  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_5  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_6  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_7  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_8  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_9  | Applied transaction identifier value |
|                     | APPLIED_USER_TRX_IDEN<br>TIFIER_VALUE_10 | Applied transaction identifier value |



---

## Transaction Account Builder

### Transaction Account Builder Overview

Use the Transaction Account Builder (TAB) to derive default accounts for their transactions using sources defined in the Accounting Methods Builder (AMB).

Derive accounting codes from the TAB and the AMB. Use the TAB to derive default accounts for transactions before they are accounted. Use the AMB to generate the accounts that appear in the accounting.

**Note:** TAB only derives default accounts for transactions. These accounts may not be the ones that appear on the subledger journal entries since these are generated by the Create Accounting program based on the application accounting definitions.

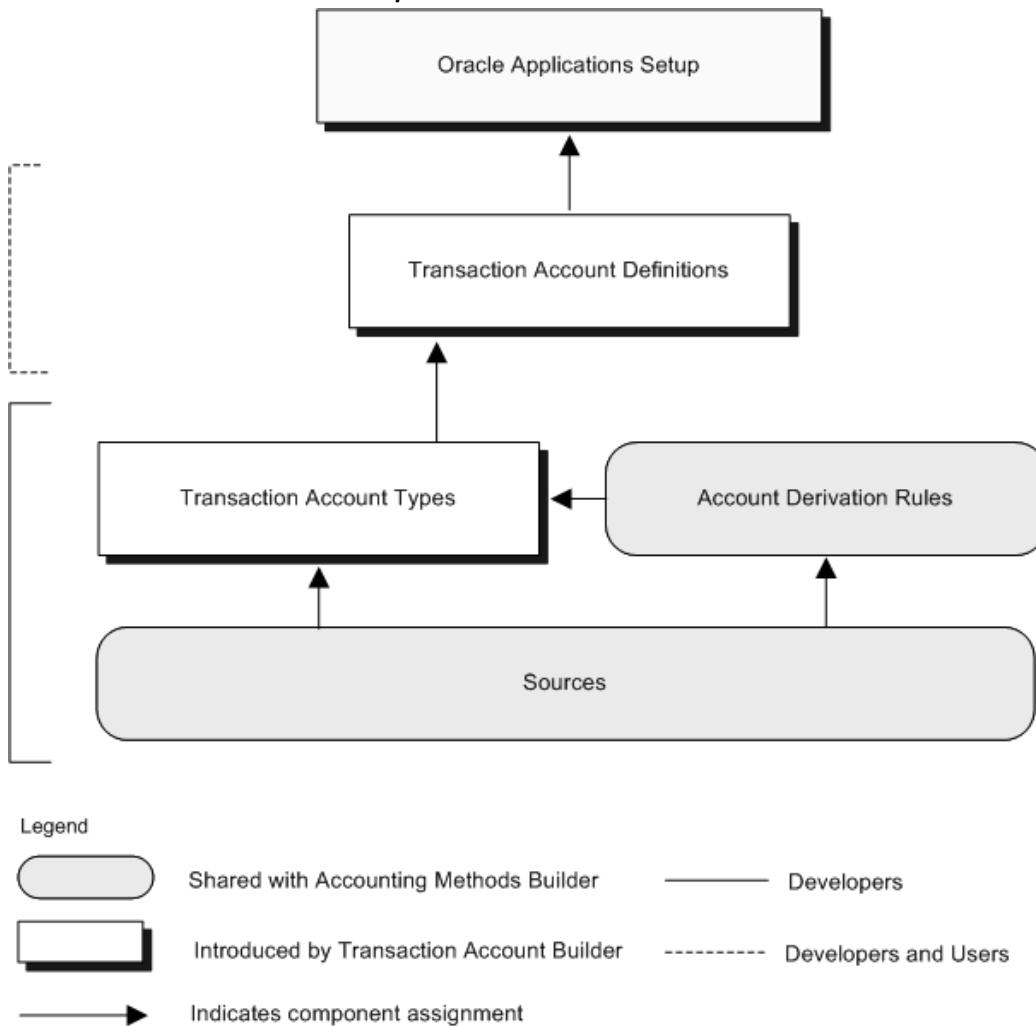
If the application does not allow modification of the default accounts for a transaction before it is accounted, you must use the AMB.

See: Accounting Methods Builder Introduction, page 2-1

### Transaction Account Components

The Transaction Account Builder Components figure below shows the components that comprise the TAB and is described in the subsequent text. Sources and account derivation rules are shared with the AMB, while transaction account types and transaction account definitions are specific to the TAB.

### Transaction Account Builder Components



Developers can assign sources to transaction account types. Developers and users can use these sources to derive accounts for a transaction account type to create a transaction account definition. For example, while the bank asset account can be used to derive the cash account for a payment, it might not be useful when deriving the invoice price variance account for a matched invoice. Consequently, the source for the bank asset account is not assigned to the transaction account type for the invoice price variance.

Examples of possible transaction account types for Receivables transactions are Revenue, Tax, Receivable, Freight, Unearned Revenue, Unbilled Receivable, and Bills Receivable. Examples of transaction account receipts are Cash, Unidentified Receipts, Unapplied Receipts, On Account Receipts, Earned Discounts, Unearned Discounts, Remitted Receipts, and Factored Receipts.

Developers and users can assign account derivation rules to transaction account types

to create a transaction account definition. These definitions can be assigned as part of the applications setup.

See: Step 2: Create Transaction Account Definitions (Optional), page 11-3

## Transaction Account Builder Setup Process

The TAB setup steps are:

- Step 1: Create Account Derivation Rules (Optional), page 11-3
- Step 2: Create Transaction Account Definitions (Optional), page 11-3
- Step 3: Assign Transaction Account Definitions to Application Setup, page 11-6
- Step 4: Enter Transactions, page 11-7
- Step 5: Using the Transaction Accounts as Default Accounts for Accounting Methods Builder, page 11-7

### Step 1: Create Account Derivation Rules (Optional)

If creating new transaction account definitions, you may need to create new account derivation rules. These rules can be of type Accounting Flexfield or of type Segment and can have a chart of accounts associated with them.

When defining account derivation rules for TAB, ensure the following:

- The transaction and the accounting chart of accounts are both null or they are the same.
- All sources used by the account derivation rule are available to the transaction account type to which it is assigned.

See: Account Derivation Rules, page 2-57

### Step 2: Create Transaction Account Definitions (Optional)

Use the transaction account definitions to assign account derivation rules to transaction account types. This determines the default account that are derived for an application.

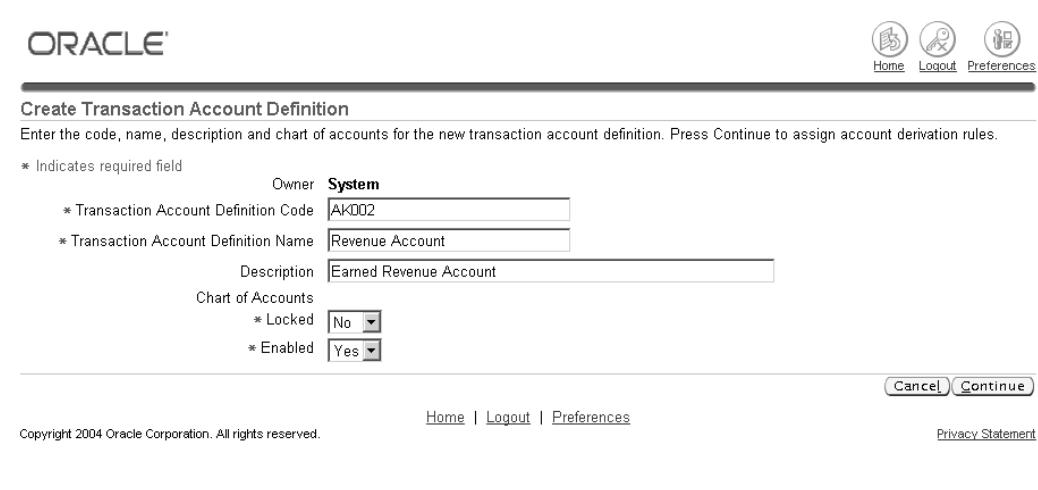
Optionally, developers can seed transaction account definitions, which must be chart of accounts independent. Use these seeded transaction account definitions and create your own by duplicating and modifying seeded definitions or by creating new ones. You can assign a chart of accounts to your definitions.

See: Account Derivation Rules, page 2-57

## Creating Transaction Account Definitions

In the Create Transaction Account Definition page, you can:

- Assign account derivation rules to transaction account types
- Compare transaction accounting definitions to determine which one meets their requirements



The screenshot shows the 'Create Transaction Account Definition' page. The form includes fields for Owner (System), Transaction Account Definition Code (AK002), Transaction Account Definition Name (Revenue Account), Description (Earned Revenue Account), Chart of Accounts (selected), Locked (No), and Enabled (Yes). Buttons for Cancel and Continue are at the bottom.

The following procedure describes selected fields.

1. In the Transaction Accounts Definition page, click **Create Transaction Account Definition**.
2. Select a Chart of Accounts to associate with the transaction account definition. Seeded transaction account definitions must be chart of accounts independent. You have the option of assigning a chart of accounts. Once a chart of accounts is associated with a transaction account definition, it cannot be changed.
3. To make the transaction account definition available for assigning as part of the application setup, select Yes in the Enabled drop-down list.
4. To assign account derivation rules, click **Continue**.
5. In the Segment field, select the segment you want to derive. If a chart of accounts is not assigned to the transaction account definition, the list of values includes All Segments and the Accounting Flexfield qualifiers. Otherwise, the list of values includes the individual segment names from the chart of accounts. If both segment and Accounting Flexfield rules are assigned, segment rules have precedence over Accounting Flexfield rules. This enables you to overwrite specific

segments of the chart of accounts. However, you are required to provide assignments for all segments of the chart of accounts. For example, you need to assign an account derivation rule to All Segments if you want to include assignments for Accounting Flexfield qualifiers; otherwise, TAB is unable to derive a complete account.

6. In the Account Derivation Rule field, enter the rule you want to use to derive the value for the segment entered in the Segment field.

The list of values for this field includes all account derivation rules that:

- Are associated with the application of the transaction account definition
- Only use sources assigned to the transaction
- Meet the conditions described in the table below:

| <b>Chart of Accounts</b> | <b>Segment</b>                  | <b>Rule Name</b>  |
|--------------------------|---------------------------------|---|
| Null                     | All Segments                    | List of values includes all account derivation rules with no transaction chart of accounts and no accounting chart of accounts and the output type is Accounting Flexfield.   |
| Null                     | Accounting Flexfield Qualifiers | List of values includes all account derivation rules with no transaction chart of accounts and no accounting chart of accounts and the output type is Segment and the Accounting Flexfield qualifier matches the Segment field. |

| Chart of Accounts | Segment      | Rule Name   |
|-------------------|--------------|---|
| Not Null          | All Segments | <p>List of values includes all account derivation rules included in case 1 plus all account derivation rules with the same transaction and accounting chart of accounts as the one of the transaction account definition and output type is Accounting Flexfield.</p>   |
| Not Null          | Segments     | <p>List of values includes all account derivation rules with no transaction chart of accounts or the same transaction chart of accounts as the transaction account definition and the following:</p> <ul style="list-style-type: none"> <li>- no accounting chart of accounts and the output type is segment qualifier and the Accounting Flexfield that matches the one for the Segment field</li> <li>- no accounting chart of accounts and the output type is the same value set as the one used by the segment</li> <li>- same accounting chart of accounts and the output type is the segment and the segment matching the Segment field.</li> </ul> |

7. To create a new account derivation rule assignment for the transaction account type, in the Assign Rule field, click the icon.

### Step 3: Assign Transaction Account Definitions to Application Setup

Refer to the application documentation for details on how to assign transaction account

definitions as part of the application setup.

## **Step 4: Enter Transactions**

Once TAB is set up for an application setup, you can create transactions for the application. Because TAB is called by the application every time a default account for a transaction is required, no action is required.

**Note:** TAB provides default accounts for transactions; therefore, you can change these accounts.

## **Step 5: Using the Transaction Accounts as Default Accounts for Accounting Methods Builder**

Use the accounts derived by TAB to generate subledger journal entries provided that the sources for these accounts are defined by the developers. To do this, complete the following steps:

1. Identify the sources that correspond to the accounts defaulted by TAB.
2. Create account derivation rules using the sources that correspond to the accounts generated by TAB.
3. Assign these account derivation rules to their transaction accounting definitions.



# A

---

## Oracle Subledger Accounting Navigation Paths

### Navigation Paths

This table lists each Subledger Accounting window and corresponding navigation path, although your system administrator may have customized your navigator.

#### *Navigation Paths*

| Window Name                         | Navigation Path  |
|-------------------------------------|--|
| Account Derivation Rules            | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Account Derivation Rules<br><br>Enter search information and click Find in the Find Account Derivation Rules window.                         |
| Account Derivation Rules Conditions | Subledger Accounting Setups – Accounting Methods Builder – Accounting Definitions Inquiry<br><br>Select the Line Inquiry Level radio button.<br><br>Select an Account Derivation Rule and click Find.<br><br>Click Conditions. |
| Accounting Attribute Assignments    | Subledger Accounting Setups – Accounting Methods Builder – Sources – Accounting Attribute Assignments  |

| Window Name                                      | Navigation Path  |
|--|--|
| Accounting Class Assignments                     | Subledger Accounting Setups – Accounting Methods Builder – Post-Accounting Programs Click Accounting Class Assignments.  |
| Accounting Definition Headers                    | Subledger Accounting Setups – Accounting Methods Builder – Accounting Definitions Inquiry<br>Select the Header Inquiry Level radio button.<br>Click Find.  |
| Accounting Definition Lines                      | Subledger Accounting Setups – Accounting Methods Builder – Accounting Definitions Inquiry<br>Select the Line Inquiry Level radio button.<br>Click Find.  |
| Accounting Event Class Options                   | Subledger Accounting Setups – Accounting Methods Builder – Events – Accounting Event Class Options   |
| Accounting Events                                | Subledger Accounting - Journal Entries Select Accounting Events tab.   |
| Accounting Setup Flow                            | Using a GL responsibility: Setup – Financials – Accounting Setup Manager – Accounting Setup Search for ledger. Click Update Accounting Options. Click the Subledger Accounting Options Update icon. Click the Update Accounting Options icon for the Subledger Accounting application. |
| Application Accounting Definitions History       | Subledger Accounting Setups – Accounting Methods Builder – Application Accounting Definition Loader – History  |
| Application Accounting Definition Merge Analysis | Subledger Accounting Setups – Accounting Methods Builder – Application Accounting Definition Loader – Merge Analysis   |

| Window Name                           | Navigation Path   |
|---------------------------------------|---|
| Application Accounting Definitions    | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Application Accounting Definitions Enter search information and click Find on the Find Application Accounting Definitions window.    |
| Copy Journal Entry Description        | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Entry Descriptions<br>Enter search information and click find in the Find Journal Entry Descriptions window.                    |
|                                       | Click Copy.   |
| Copy Journal Lines Definition         | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Journal Line Definitions Enter search information and click Find in the Find Journal Line Definitions window. Click Copy Definition. |
| Create Subledger Journal Entry        | Inquiry - Journal Entries. Select Subledger Journal Entries tab. Click Create Journal Entry.  |
| Create Transaction Account Definition | Subledger Accounting Setups – Transaction Account Builder – Test Transaction Account Builder Click Create Transaction Account Definition  |
| Custom Sources                        | Subledger Accounting Setups – Accounting Methods Builder – Sources – Custom Sources<br>Enter search information and click Find on Find Custom Sources window.   |
| Delete Subledger Journal Entries      | Inquiry - Journal Entries. Select Subledger Journal Entries tab. Search for ledger. Check Select box for ledger to be deleted. Click Delete.  |
| Duplicate Subledger Journal Entries   | Inquiry - Journal Entries. Select Subledger Journal Entries tab. Search for ledger. Check Select box for ledger to be duplicated. Click Duplicate.  |

| Window Name                             | Navigation Path  |
|---|--|
| Entities                                | Subledger Accounting Setups – Accounting Methods Builder – Events – Event Model  |
| Event Class Predecessors                | Subledger Accounting Setups – Accounting Methods Builder – Events – Event Model Click Event Classes. Click Predecessors. |
| Event Classes and Types                 | Subledger Accounting Setups – Accounting Methods Builder – Events – Event Model Click Event Classes.                     |
| Find Account Derivation Rules           | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Account Derivation Rules               |
| Find Accounting Definitions             | Subledger Accounting Setups – Accounting Methods Builder – Accounting Definitions Inquiry                                |
| Find Application Accounting Definitions | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Application Accounting Definitions  |
| Find Custom Sources                     | Subledger Accounting Setups – Accounting Methods Builder – Sources – Custom Sources                                      |
| Find Journal Entry Descriptions         | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Entry Descriptions             |
| Find Journal Lines Definitions          | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Journal Lines Definition            |
| Find Journal Line Types                 | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Line Types                     |
| Find Mapping Sets                       | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Mapping Sets                           |

| Window Name                          | Navigation Path   |
|--------------------------------------|---|
| Find Subledger Accounting Methods    | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Subledger Accounting Methods       |
| Header Assignments                   | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Application Accounting Definitions |
|                                      | Enter search information and click Find on the Find Application Accounting Definitions window.                          |
| Click Header Assignments.            |   |
| Journal Entry Description Conditions | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Entry Descriptions            |
|                                      | Enter search information and click find in the Find Journal Entry Descriptions window.                                  |
| Click Conditions.                    |   |
| Journal Entry Description Details    | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Entry Descriptions            |
|                                      | Enter search information and click find in the Find Journal Entry Descriptions window.                                  |
| Click Details.                       |   |
| Journal Entry Descriptions           | Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Entry Descriptions            |
|                                      | Enter search information and click find in the Find Journal Entry Descriptions window.                                  |
| Click Descriptions.                  |   |
| Journal Line Definitions             | Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Journal Line Definitions           |
|                                      | Enter search information and click Find in the Find Journal Line Definitions window.                                    |
| Click Line Definitions.              |   |

| Window Name                                    | Navigation Path  |
|--|--|
| Journal Line Type Conditions                   | <p>Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Line Types</p> <p>Enter search information and click Find in the Find Journal Line Types window.</p> <p>Click Conditions.</p>                       |
| Journal Line Types                             | <p>Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Line Types</p> <p>Enter search information and click Find in the Find Journal Line Types window.</p>  |
| Journal Line Accounting Attributes Assignments | <p>Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Journal Line Types</p> <p>Enter search information and click Find in the Find Journal Line Types window.</p> <p>Click Accounting Attribute Assignments.</p> |
| Journal Lines Definitions                      | <p>Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Journal Lines Definitions</p>  |
| Mapping Sets                                   | <p>Subledger Accounting Setups – Accounting Methods Builder – Journal Entry Setups – Mapping Sets</p> <p>Enter search information and click Find in the Find Mapping Sets window.</p>  |
| Multiperiod Accounting                         | <p>Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Journal Lines Definitions</p> <p>Click Multiperiod Accounting.</p>   |
| Open Account Balances Listing Definitions      | <p>Subledger Accounting Setups – Accounting Methods Builder – Open Account Balances Listing Definitions</p>  |

| Window Name                     | Navigation Path  |
|---------------------------------|--|
| Post-Accounting Programs        | Subledger Accounting Setups – Accounting Methods Builder – Post-Accounting Programs  |
| Process Categories              | Subledger Accounting Setups – Accounting Methods Builder – Events – Process Categories   |
| Reverse Subledger Journal Entry | <p>Inquiry - Journal Entries</p> <p>Click Subledger Journal Entries tab.</p> <p>Search for ledger.</p> <p>Click Reverse.</p>   |
| Source Assignments              | Subledger Accounting Setups – Accounting Methods Builder – Sources – Source Assignments  |
| Sources                         | Subledger Accounting Setups – Accounting Methods Builder – Sources – Sources   |
| Subledger Accounting Methods    | <p>Subledger Accounting Setups – Accounting Methods Builder – Methods and Definitions – Subledger Accounting Methods</p> <p>Enter search information and click Find on the Find Subledger Accounting Methods window.</p> |
| Subledger Applications          | Subledger Accounting Setups – Subledger Applications   |
| Subledger Journal Entries       | Subledger Accounting - Journal Entries   |
| Subledger Journal Entry Lines   | <p>Subledger Accounting - Journal Entries</p> <p>Select Subledger Journal Entries tab.</p> <p>Select Lines tab.</p>  |
| Supporting Reference Balances   | Subledger Accounting - Supporting Reference Balances   |

| Window Name                      | Navigation Path   |
|----------------------------------|---|
| Supporting References            | Subledger Accounting Setup - Accounting Methods Builder - Journal Entry Setups - Supporting References                                    |
| System Transaction Identifiers   | Subledger Accounting Setups – Accounting Methods Builder – Events – Event Model<br><br>Click Identifiers.                                 |
| Test Transaction Account Builder | Subledger Accounting Setups – Transaction Account Builder – Test Transaction Account Builder  |
| Transaction Account Definitions  | Subledger Accounting Setups – Transaction Account Builder – Transaction Account Definitions   |
| Transaction Account Types        | Subledger Accounting Setups – Transaction Account Builder – Transaction Account Types   |
| Update Subledger Journal Entry   | Inquiry - Journal Entries<br><br>Select Subledger Journal Entries tab.<br><br>Search for ledger.<br><br>Select line.<br><br>Click Update. |

# B

---

## Oracle Subledger Accounting Profile Options and Profile Option Categories

### Profile Options and Profile Option Categories Overview

During implementation, set a value for each Subledger Accounting profile option to specify how Subledger Accounting controls access to and processes data.

See: Setting User Profile Options, *Oracle Applications System Administrator's Guide - Maintenance*

Profile options are grouped into one or more profile option categories enabling you to view only the profile options that pertain to your application or function.

#### Subledger Accounting Categories

- Accounting Methods Builder, page B-3
- Debug, page B-3
- Deployment, page B-6
- Security, page B-7

#### Subledger Accounting Profile Options

- SLA: Accounting Methods Builder Context, page B-3
- SLA: Additional Data Access Set, page B-8
- SLA: Allow Reports Journal Source Override, page B-8
- SLA: Disable Journal Import, page B-7
- SLA: Display Timestamp on Trace, page B-5

- SLA: Enable Diagnostics, page B-5
- SLA: Enable PL/SQL Profiler, page B-5
- SLA: Enable Trace, page B-5
- SLA: Enable SQL Trace, page B-5
- SLA: Enable SRS Log/Output, page B-5
- SLA: Flush Trace File, page B-6
- SLA: Oracle Forms Trace Mode, page B-6
- SLA: SRS Trace Mode, page B-6
- SLA: Trace File Override Directory, page B-6
- SLA: Trace File Override Filename, page B-6
- SLA: Trace Level, page B-6
- SLA: Trace Timeout, page B-6
- SLA: Enable Data Access Security in Subledgers, page B-8
- SLA: Enable Subledger Transaction Security in GL, page B-9

## Profile Option Category and Profile Option Descriptions

This section describes the profile options by category.

The tables in this section provide profile option information as follows:

- The Default column displays the default profile option value in *italics* or No Default if none exists.
- The User Access column indicates whether you can view or update the profile option.
- The System Administration: User, Responsibility, Application, and Site columns indicate at which levels the system administrator can update these profile options.

The key for these tables is:

- Update: You can update the profile option.
- View Only: You can view the profile option but cannot change it.

- No Access: You cannot view or change the profile option value.

## Accounting Methods Builder Category

This category includes profile options related to the Accounting Methods Builder (AMB).

### ***Accounting Methods Builder Category***

| Profile Option                                    | Default        | User Access | System Administration : User | System Administration : Responsibility | System Administration: Application | System Administration : Site |
|---|----------------|-------------|------------------------------|--|------------------------------------|------------------------------|
| SLA: Accounting Methods Builder Context, page B-3 | <i>Default</i> | View Only   | Update                       | Update                                 | Update                             | Update                       |

### **SLA: Accounting Methods Builder Context**

This profile option controls the value for an internal context code associated with each application accounting definition and each of its assigned components, such as journal line types and account derivation rules.

Most AMB components are striped by a context. Assign a different value for the SLA: Accounting Methods Builder Context profile option to work on your own copy of components and perform changes.

Define additional AMB Context profile values as a Subledger Accounting lookup, XLA\_AMB\_CONTEXT\_TYPE.

## Debug Category

This category includes profile options that enable debugging and diagnostic features.

**Debug Category**

| Profile Option                               | Default    | User Access | System Administration: User | System Administration: Responsibility | System Administration: Application | System Administration: Site |
|--|------------|-------------|-----------------------------|---------------------------------------|------------------------------------|-----------------------------|
| SLA: Display Timestamp on Trace, page B-5    | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Enable Diagnostics, page B-5            | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Enable PL/SQL Profiler, page B-5        | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Enable Trace, page B-5                  | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Enable SQL Trace, page B-5              | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Enable SRS Log/Output, page B-5         | Yes        | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Flush Trace File, page B-6              | No         | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Oracle Forms Trace Mode, page B-6       | File       | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: SRS Trace Mode, page B-6                | Tracer     | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Trace File Override Directory, page B-6 | No Default | View Only   | Update                      | Update                                | Update                             | Update                      |

| Profile Option                                 | Default    | User Access | System Administration: User | System Administration: Responsibility | System Administration: Application | System Administration: Site |
|--|------------|-------------|-----------------------------|---------------------------------------|------------------------------------|-----------------------------|
| SLA: Trace File Override<br>Filename, page B-6 | No Default | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Trace Level, page B-6                     | 100        | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Trace Timeout, page B-6                   | 600        | View Only   | Update                      | Update                                | Update                             | Update                      |

### **SLA: Display Timestamp on Trace**

This profile option displays timestamps on the trace.

### **SLA: Enable Diagnostics**

This profile option controls whether diagnostic information is gathered by the Create Accounting program.

If the FND: Debug Log Enabled profile option is enabled, then the diagnostic framework information is also available in the Oracle Applications system logs page.

### **SLA: Enable PL/SQL Profiler**

This profile option enables the PL/SQL profiler.

### **SLA: Enable Trace**

This profile option enables the Subledger Accounting trace mode.

### **SLA: Enable SQL Trace**

This profile option enables the SQL trace. The possible values are Yes or No.

### **SLA: Enable SRS Log/Output**

This profile option controls whether concurrent programs will generate a log or output file.

### **SLA: Flush Trace File**

This profile option controls whether the trace information is sent to the trace file immediately.

### **SLA: Oracle Forms Trace Mode**

This profile option indicates where the debug information provided by the forms is printed. The possible values are AF Log and File.

### **SLA: SRS Trace Mode**

This profile option is set by Subledger Accounting concurrent programs. It indicates where the debug information provided by the concurrent programs is printed. The possible values are AF File and Logfile.

### **SLA: Trace File Override Directory**

This profile option designates the directory for the trace file.

### **SLA: Trace File Override Filename**

This profile option designates the trace file name.

### **SLA: Trace Level**

This profile option controls the level of detail for the trace.

### **SLA: Trace Timeout**

This profile option controls the time a program should wait for acknowledgement from the trace engine.

## **Deployment Category**

This category includes profile options set up to run the application.

### ***Deployment Category***

| Profile Option                        | Default | User Access | System Administration: User | System Administration: Responsibility | System Administration: Application | System Administration: Site |
|---------------------------------------|---------|-------------|-----------------------------|---------------------------------------|------------------------------------|-----------------------------|
| SLA: Disable Journal Import, page B-7 | No      | No Access   | View Only                   | View Only                             | View Only                          | Update                      |

| Profile Option                                     | Default   | User Access | System Administration: User | System Administration: Responsibility | System Administration: Application | System Administration: Site |
|--|-----------|-------------|-----------------------------|---------------------------------------|------------------------------------|-----------------------------|
| SLA: Initial Date for Historical Upgrade, page B-7 | No Access | View Only   | View Only                   | View Only                             | Update                             |                             |

### SLA: Disable Journal Import

This profile option controls whether subledger journal entries are imported to General Ledger.

See: Create Accounting Program, page 3-1

### SLA: Initial Date for Historical Upgrade

This profile option initiates the Historical Upgrade on the specified date.

## Security Category

This category includes profile options related to security.

### Security Category

| Profile Option   | Default    | User Access | System Administration: User | System Administration: Responsibility | System Administration: Application | System Administration: Site |
|--|------------|-------------|-----------------------------|---------------------------------------|------------------------------------|-----------------------------|
| SLA: Additional Data Access Set, page B-8                | No Default | View Only   | Update                      | Update                                | Update                             | Update                      |
| SLA: Allow Reports Journal Source Override, page B-8     | No         | View Only   | Update                      | No Access                             | Update                             | Update                      |
| SLA: Enable Data Access Security in Subledgers, page B-8 | No         | View Only   | No Access                   | Update                                | Update                             | Update                      |

| Profile Option   | Default | User Access | System Administration : User | System Administration : Responsibility | System Administration : Application | System Administration: Site |
|--|---------|-------------|------------------------------|--|-------------------------------------|-----------------------------|
| SLA: Enable Subledger Transaction Security in GL, page B-9 | No      | View Only   | No Access                    | Update                                 | Update                              | Update                      |

### SLA: Additional Data Access Set

The SLA: Additional Data Access Set profile option, in conjunction with the GL: Data Access Set profile option, controls which ledgers and balancing or management segment values you can access when logging onto a responsibility. If SLA: Enable Data Access Security in Subledgers is enabled for the responsibility, you have access only to the ledgers and balancing or management segment values included in the data access sets assigned to the SLA: Additional Data Access Set and GL: Data Access Set profile options.

### SLA: Allow Reports Journal Source Override

This profile option applies only to the following reports:

- Open Account Balances Listing
- Third Party Balances Report

Enable this option to change the Journal Source parameter during report submission. If the option is set to No, then you cannot change the value defaulted during report submission.

### SLA: Enable Data Access Security in Subledgers

This profile option determines whether the General Ledger Access Set security mechanism is applied for a subledger application responsibility when viewing, reporting, or creating subledger journal entries associated with a given ledger. The General Ledger Access Set security mechanism is always applied for responsibilities associated with the General Ledger application.

The profile option enables you to combine data access security with subledger transaction security and therefore control access to subledger journal entries depending on the ledger to which they belong. For example, you can implement a Multi-Org Security Profile that allows you to create Oracle Receivables Invoices for two different operating units each associated with different ledgers but restrict drill-down from the subledger transaction to the associated subledger journal entry based upon the

destination ledger contained in the Access Set.

### **SLA: Enable Subledger Transaction Security in GL**

Use this profile option to combine subledger transactions security with data access security for General Ledger responsibilities when drilling down to multi-organization enabled subledger application. Transaction security in the respective subledger application is always applied when drilling down from subledger transactions to subledger journal entries.



---

## Subledger Journal Entry Definition

### Subledger Journal Entry Definition Overview

The goal of Subledger Accounting is to generate journal entries for transactions that occur in subledgers. For users integrating subledger accounting into their applications, it is critical to have a clear and common definition of a subledger journal entry. A common definition ensures the validity of all journal entries generated by Subledger Accounting.

This chapter provides detailed information on the elements that make up a subledger journal entry and the validations enforced by Subledger Accounting. Where applicable, information is also provided on the default values stored in subledger accounting tables.

### High Level Definition of a Subledger Journal Entry

A valid subledger journal entry must satisfy all the rules and validations listed below. If any of these rules and validations are not applied successfully, the journal entry has an invalid status.

- The subledger journal entry must have one header.

See: Subledger Journal Entry Header Validation, page C-2

- Actual and encumbrance journal entries must have at least two journal entry lines.

See: Subledger Journal Entry Line Validation, page C-9

- Actual and encumbrance journal entries must be balanced in both entered and ledger currencies. A journal entry need not be balanced if it is a STAT currency entry.

In addition, actual and encumbrance entries must be balanced by balancing segment.

See: Subledger Journal Entry Balancing Rules, page C-14.

- Encumbrance journal entries can only be created for ledgers that are enabled for encumbrance accounting.
- The subledger journal entry must have valid monetary amounts.  
See: Validation of Monetary Amounts, page C-16
- The actual journal entry must be created in an open or future open GL period. Similarly, a budget journal entry must be created in an open budget year and an encumbrance journal entry created in an open encumbrance year.
- All the lines on a subledger journal entry must have valid accounts.
- Budget journal entries are not created for reporting currencies and secondary ledgers.

## Subledger Journal Entry Header Validation

Header information for subledger journal entries is stored in the XLA\_AE\_HEADERS table.

For information on the XLA\_AE\_HEADERS table, see the Oracle Subledger Accounting eTRM.

This section provides detailed information on the validation rules for all subledger journal entry header attributes. Where applicable, the rules also describe whether a journal entry header is created if the validation fails and what values are stored in case of errors.

### 1. Header Identifier (AE\_HEADER\_ID)

The header identifier uniquely identifies the subledger journal entry header. A unique header identifier is generated using the sequence XLA\_AE\_HEADERS\_S.

### 2. Application (APPLICATION\_ID)

A subledger journal entry belongs to one and only one subledger application. The application identifier is the standard identifier for each Oracle e-Business Suite application or custom application.

### 3. Ledger (LEDGER\_ID)

A subledger journal entry belongs to one and only one ledger. The ledger identifier is the standard identifier assigned to the ledger by the Accounting Setup Manager (ASM).

#### 4. Transaction Entity Identifier (ENTITY\_ID)

The transaction entity identifier identifies the source transaction associated with a subledger journal entry. It is always populated.

The following rule applies:

- It exists in the XLA\_TRANSACTION\_ENTITIES table.

#### 5. Event Identifier and Event Type Code (EVENT\_ID, EVENT\_TYPE\_CODE)

A subledger journal entry is attached to one and only one accounting event. However, an accounting event can be used to generate multiple subledger journal entries when using multiple accounting representations.

The event identifier (EVENT\_ID) is the standard identifier for each event in Subledger Accounting. The event type code (EVENT\_TYPE\_CODE) is also stored in the journal entry header.

The following rules apply:

- A journal entry header is not be created if an event type is not enabled.
- A manual subledger journal entry has an event type of Manual automatically assigned by Subledger Accounting.

#### 6. GL Date (ACCOUNTING\_DATE)

The GL date determines the accounting period (PERIOD\_NAME) of the subledger journal entry and when the entry appears in fiscal reports and queries.

The following rules apply:

- For actual journal entries, the GL date must be in an open or future enterable GL period.
- For an encumbrance journal entry, the GL date must be in an open encumbrance year.
- For a budget journal entry, the GL date must in an open budget year.
- An entry is marked Invalid if it fails period validations. The Accounting Program continues to create lines even if the journal entry fails period validations.
- For a multiperiod journal entry, the GL date may fall in a never opened period. In this case, the Transfer to GL process does not transfer the journal entry to General Ledger until the GL period status is changed to either Open or Future Enterable.
- The GL date has no time component. It is not subject to considerations related to

timezone differences.

## 7. Reference Date (REFERENCE\_DATE)

The reference date can be used as a basis for the reporting sequence. It is an optional attribute. When a subledger journal entry is created and completed, the reference date must belong to an open or future open period in General Ledger.

The following rules apply:

- When a subledger journal entry is completed, the reference date must be in an open or future open accounting period.
- Reference date is only stored for actual manual journal entries.
- The reference date has no time component. It is not subject to considerations related to timezone differences.

## 8. Balance Type Code (BALANCE\_TYPE\_CODE)

The balance type of a subledger journal entry has one of the following values:

- Actual
- Budget
- Encumbrance

The following rules apply:

- For actual entries, the budget version identifier and encumbrance type identifier are null.
- For budget entries, the budget version identifier is not null, and the encumbrance type identifier is null.
- For encumbrance entries, the encumbrance type identifier is not null, and the budget version identifier is null.

## 9. Journal Entry Category Name (JE\_CATEGORY\_NAME)

This is the name of the General Ledger journal category.

The following rules apply:

- For manual and upgrade journal entries, users must provide the journal category.
- For standard and multiperiod journal entries, the journal category is derived from

the category specified for an event class in the Subledger Accounting Options page of the Accounting Setups.

## 10. GL Transfer Status Code (GL\_TRANSFER\_STATUS\_CODE)

The GL transfer status code of the subledger journal entry has one of the following values:

- Not transferred
- Selected to transfer
- Transferred

When a subledger journal entry is created and completed, the value of this column is Not transferred. The Transfer to GL process updates this column when the subledger journal entry is transferred to General Ledger.

## 11. Subledger Journal Entry Status Code (ACCOUNTING\_ENTRY\_STATUS\_CODE)

A subledger journal entry can have a status of Draft, Final, Incomplete, Invalid or Invalid Related Entry.

**Draft:** The entry status is set to Draft when the following conditions are met:

- The subledger journal entry is accounted in draft mode.
- The subledger journal entry is valid.
- The originating event has no errors. (All other subledger journal entries associated with the same event are valid.)
- All prior events for the same document have been accounted.

**Final:** The entry status is set to Final when the following conditions are met:

- The subledger journal entry is accounted in final mode.
- The subledger journal entry is valid.
- The originating event has no errors. (All other subledger journal entries associated with the same event are valid.)
- All prior events for the same document have been accounted.

**Incomplete:** The entry status is set to Incomplete when the following condition is met:

- The subledger journal entry is incomplete and subject to further changes. This status is applicable to manual and multiperiod journal entries.

**Invalid:** The entry status is set to Invalid when the following condition is met:

- The subledger journal entry has errors.

**Invalid Related Entry:** The entry status is set to Invalid Related Entry when the following condition is met:

- The subledger journal entry is valid, but one or more of the other subledger journal entries associated with the same accounting event are invalid.

## 12. Subledger Journal Entry Type Code (ACCOUNTING\_ENTRY\_TYPE\_CODE)

The type of a subledger journal entry is one of the following:

- Standard
- Upgrade
- Manual
- Multiperiod

## 13. Application Accounting Definition Type Code, Application Accounting Definition Code, Application Accounting Definition Version (PRODUCT\_RULE\_TYPE\_CODE, PRODUCT\_RULE\_CODE, PRODUCT\_RULE\_VERSION)

The application accounting definition type code, code and version store information about the application accounting definition used by the Accounting Program to create the journal entry.

The following rule applies to these identifiers:

- They are not populated for manual, upgrade, or third party merge entries.

## 14. Description (DESCRIPTION)

This is the description of the subledger journal entry header. For standard, upgrade, and third party merge journal entries, the description is always in the language of the ledger.

The following rule applies to a subledger journal entry description:

- A header description is required for manual journal entries. It is optional for other journal entry types.

## 15. Document Category (DOC\_CATEGORY\_CODE)

This is the identifier of the document sequence category used to derive the document

sequence of the subledger transaction.

## **16. Document Sequence Identifier (DOC\_SEQUENCE\_ID) and Value (DOC\_SEQUENCE\_VALUE)**

These are the sequence identifiers and values that correspond to a valid document sequence at the time the journal entry is created.

## **17. Accounting Batch Identifier (ACCOUNTING\_BATCH\_ID)**

This is the identifier of the batch in which the subledger journal entry has been created.

The following rule applies:

- This column is null for manual, third party merge, and upgrade subledger journal entries.

## **18. Accounting Sequence Version, Identifier and Value (COMPLETION\_ACCT\_SEQ\_VERSION\_ID, COMPLETION\_ACCT\_SEQ\_ASSIGN\_ID, COMPLETION\_ACCT\_SEQ\_VALUE)**

These are the sequence versions, identifiers, and values assigned to the completed subledger journal entry.

## **19. Period Close Sequence Version, Identifier, and Value (CLOSE\_ACCT\_SEQ\_VERSION\_ID, CLOSE\_ACCT\_SEQ\_ASSIGN\_ID, CLOSE\_ACCT\_SEQ\_VALUE)**

These are the sequence versions, identifiers, and values assigned to the subledger journal entry at period close.

## **20. Budget Version Identifier (BUDGET\_VERSION\_ID)**

This is the identifier of the budget version.

The following rules apply:

- It is populated only for budget entries.
- The budget must exist, it must not be frozen or inactive, and it must be associated with a ledger.

No budget subledger journal entries are created if it is created with a status or Error.

## **21. Encumbrance Type Identifier (ENCUMBRANCE\_TYPE\_ID)**

This is the identifier for the encumbrance type.

The following rules apply:

- It is populated only for encumbrance entries.
- The encumbrance type must exist and be enabled.

No encumbrance subledger journal entries are created if any of the above validations fail.

## 22. Funds Status Code (FUNDS\_STATUS\_CODE)

The funds status code of the subledger journal entry is one of the following:

- Passed
- Failed
- In Progress

This is an optional column. The following rule applies to the funds status code:

- If funds checking is enabled for a ledger, then there must be available funds for each line. If this condition is not met, the status of the subledger journal entry is set to Invalid and the funds status is set to Failed.

Users are not allowed to complete a manual subledger journal entry with a funds status Failed.

## 23. Budgetary Control Packet Identifier (PACKET\_ID)

This is the identifier of the budgetary control packet in the GL\_BC\_PACKETS table. It is used to implement budgetary control.

The following rule applies:

- It is populated only for manual journal entries.

## 24. Group Identifier (GROUP\_ID)

The group identifier is used by the journal import to identify journal import batches.

## 25. Completion Date (COMPLETED\_DATE)

The completion date is the date a valid journal entry is accounted for in final mode.

The completion date has a time component. It is subject to considerations related to timezone differences.

## Subledger Journal Entry Line Validation

Line information for subledger journal entries is stored in the XLA\_AE\_LINES table.

For information on the XLA\_AE\_LINES table, see the Oracle Subledger Accounting eTRM.

This section provides detailed information on the validation rules for all subledger journal entry line attributes.

### 1. Header Identifier (AE\_HEADER\_ID)

This is the identifier of the subledger journal entry header. All lines of a given subledger journal entry share the same header identifier.

### 2. Line Number (AE\_LINE\_NUM)

This is a sequential number that identifies each line within a subledger journal entry. AE\_HEADER\_ID and AE\_LINE\_NUM constitute the unique key for a subledger journal entry line. There is a one-to-many relationship between a subledger journal entry header and subledger journal entry lines.

### 3. Displayed Line Number (DISPLAYED\_LINE\_NUM)

This is a sequential number that identifies each subledger journal entry line within a subledger journal entry. It is an updateable value when creating a manual subledger journal entry.

### 4. Account Code Combination (CODE\_COMBINATION\_ID)

This is the code combination identifier that identifies the General Ledger account to which the subledger journal entry line is posted.

The following rules apply:

- It must be a valid code combination. For performance reasons, the following validation is performed:
  - The Accounting Program checks whether the code combination exists in the table GL\_CODE\_COMBINATIONS. The Accounting Program assumes that compliance with General Ledger cross validation rules has been met by the base product.
  - Instead of deriving the account from sources in the transaction objects, the code combination can also be derived from a constant value specified in an account derivation rule. In this case, it is also assumed that no General Ledger cross validation rules are violated.

- If dynamic insertion is turned on and the code combination does not exist, then a new combination is created. In this case, Subledger Accounting calls the corresponding Applications Object Library APIs, which perform all needed validations including General Ledger cross validation rules.
- It must be active as of the GL date.
- If disabled and a substitute account is defined in General Ledger, Subledger Accounting uses the substitute account. If the substitute account is invalid and a suspense account is defined, Subledger Accounting uses the suspense account. An error message is displayed if a valid suspense account is not available.
- It must not be a summary account.
- For actual and encumbrance type entries, the Allow Posting option for the account code combination must be enabled.
- For budget entries, the Allow Budgeting option for the account code combination must be enabled.
- When using a suspense account, the suspense option must be enabled. (Suspense accounts are derived from the ledger setup in General Ledger.)
- If it is a third party control account, then the party type on subledger application registration must match the party type on the line.
- If it is a third party control account, then the party type on the line must match the party type on the account.
- The user must have write access to the ledger; ledger or balancing segment value; or ledger or management segment value.
- If the account code combination does not exist, a -1 must be stored in the CODE\_COMBINATION\_ID column.

## 5. GL Transfer Mode Code (GL\_TRANSFER\_MODE\_CODE)

This lookup code determines how subledger journal entry lines are transferred to General Ledger. The following rules apply:

- It has a value of Detail or Summary.
- For standard and deferred journal entry lines, the attribute value is derived from the journal line type setup in the AMB.
- For manual journal entries, it is derived from the Subledger Accounting options entered for the ledger in the ASM.

## 6. General Ledger Link Identifiers (GL\_SL\_LINK\_TABLE, GL\_SL\_LINK\_ID)

These identifiers link subledger journal entry lines with GL journal entry lines.

- For entries created by Subledger Accounting, GL\_SL\_LINK\_TABLE always has the value XJAJEL. For upgraded entries, values map to values transferred to General Ledger in Release 11*i*.
- GL\_SL\_LINK\_ID is null when the subledger journal entry is completed. This column is populated by the Transfer to GL process when the subledger journal entry is transferred to the General Ledger.

## 7. Accounting Class Code (ACCOUNTING\_CLASS\_CODE)

This lookup code indicates the accounting class for a subledger journal line. The following rules apply:

- The accounting class is derived from the journal line type.
- It is not populated for third party merge subledger entries.

## 8. Party Identifier (PARTY\_ID)

This party identifier is associated with the subledger transaction that generates the journal entry. As some subledger products do not have third parties, the column is optional.

The following rules apply:

- If party site information is provided, it must be valid.
- If the party identifier is provided, it must be a valid party identifier.
- If party type information is provided, then party information must also be provided and vice versa. If party type information is not provided then party information must be null and vice versa.
- If the journal entry line account is a third party control account, then party information must be provided.

The party identifier is null if any of the above validations fail. The journal entry is marked as Invalid if party information is not provided when required.

## 9. Party Site Identifier (PARTY\_SITE\_ID)

This is the party site identifier of the subledger transaction associated with the event. As some subledger products do not have third parties, this column is optional.

The following rules:

- If party site information is provided, then party information must also be provided.
- The party site must be null if any of the other validations fail.

## 10. Party Type Code (PARTY\_TYPE\_CODE)

This column identifies the type of third party associated with the subledger transaction. Currently, there are two valid values for this field as follows:

- Customers
- Suppliers

Since not all subledger products implement Oracle Trading Community Architecture at the same time, third party identifiers can be identical. It is therefore necessary to identify them based on third party type code.

## 11. Entered Amount (ENTERED\_DR, ENTERED\_CR)

The following rules apply:

- It must be in the currency of the transaction.
- It must be either a debit or a credit, but not both.
- It must be on the same side as that of the accounted amount.
- It must follow the precision of the entered currency.
- If the entered currency is the same as the ledger currency and the accounting class of the journal line is not Rounding, then entered and accounted amounts must be the same.

## 12. Accounted Amount (ACCOUNTED\_DR, ACCOUNTED\_CR)

The following rules apply to the accounted amount:

- It must be in the functional currency of the ledger.
- It must be either a debit or a credit but not both.
- It must be on the same side as that of the entered amount.
- It must follow the precision of the ledger currency.
- If the entered currency is the same as the ledger currency and the accounting class

of the journal line is not Rounding, then entered and accounted amounts must be the same.

### **13. Statistical Amount (STATISTICAL\_AMOUNT)**

This is the statistical amount of the subledger journal entry line. It is an optional column.

The following rule applies to statistical amount:

- If the profile option Journals: Mix Statistical and Monetary is set to Yes, then statistical and monetary amounts can be entered on the same journal entry.
- Statistical amount must be null if the entered currency is STAT.
- A journal entry is invalid if it contains both statistical and monetary currency lines.
- STAT currency is only used for Actual entries.

### **14. Currency Code (CURRENCY\_CODE)**

This is the entered currency. It is always populated. When the entered currency is the same as the ledger currency, this column is populated with the ledger currency.

A subledger journal entry can have lines with different entered currencies. In Payables, a payment can be made against invoices with different invoice currencies. In this case, the currency code can be different within a subledger journal entry.

The following rules apply:

- It must be provided for each journal entry line.
- If the currency code is STAT, then the entered amount is the same as the accounted amount. The statistical amount (STATISTICAL\_AMOUNT) is null.
- STAT currency and other currencies cannot coexist in a single subledger journal entry.
- STAT journals need not be balanced.
- STAT currency lines are not created for budget and encumbrance journals.

### **15. Currency Conversion Type, Date, and Rate (CURRENCY\_CONVERSION\_TYPE, CURRENCY\_CONVERSION\_DATE, CURRENCY\_CONVERSION\_RATE)**

This is the conversion rate types, conversion dates, and conversion rates associated with the journal entry line.

The following rules apply:

- When the entered and ledger currencies are the same, and the conversion rate and the conversion rate type are populated, then the conversion rate and conversion rate type values are ignored and not used.
- When the entered currency is different from ledger currency, the conversion information is not required to be stored in a journal entry line. However, if conversion information is populated, the following conditions are considered errors:
  - Conversion type is User and conversion rate is blank.
  - Conversion type is not User and conversion date is blank.

## 16. Reconciliation Reference (JGZZ\_RECON\_REF)

This is the reconciliation reference value. This column is optional and no validation is applied.

## 17. Control Balance (CONTROL\_BALANCE\_FLAG)

This flag indicates whether the subledger journal entry line contributes to the control account balance calculation. The Control Balance flag can have the following values:

- N: not eligible for balance calculation
- P: balance calculation pending
- Y: balance calculation complete

## 18. Analytical Balance (ANALYTICAL\_BALANCE\_TYPE)

This flag indicates whether the subledger journal entry line contributes to the supporting reference balance calculation. The Analytical Balance flag can have the following values:

- N: not eligible for balance calculation
- P: balance calculation pending
- Y: balance calculation complete

## Subledger Journal Entry Balancing Rules

**Note:** Balancing rules apply to actual and encumbrance journal entries only. They do not apply to budget entries which can be unbalanced or

even one-sided. Actual and encumbrance journal entries must satisfy all three balancing conditions.

The rules described in this section apply to any given subledger journal entry as a whole. Therefore, these conditions are not evaluated until the journal entry header and all the journal entry lines are generated.

## Balanced by Ledger Currency

A subledger journal entry must be balanced in the currency of the ledger. This means that the sum of the accounted debit amounts (ACCOUNTED\_DR) must equal the sum of the accounted credit amounts (ACCOUNTED\_CR). Since one journal entry line cannot have both debit and credit amounts, a journal entry must have at least two journal entry lines, one with a debit amount and another with a credit amount.

The Accounting Program balances subledger journal entries by creating balancing entries to the Ledger Currency Balancing Account specified in the ledger definition if the journal entry is not balanced by ledger currency. Select the Balance by Ledger Currency option to enable the assignment of a Ledger Currency Balancing Account. If Balance by Ledger Currency is not enabled, the Accounting Program raises an exception when a journal entry is out of balance by ledger currency.

The Entered Currency Balancing Account is used by Subledger Accounting to balance foreign currency subledger journals by the entered currency and balancing segment value. If you remove the balance account information from accounting setups, Subledger Accounting cannot derive the CCID for Balance by currency rows.

## Balanced by Entered Currency

A subledger journal entry must be balanced in the entered currency. If a subledger journal entry has lines with different entered currencies, the subledger journal entry must be balanced for each currency. This means that for each entered currency (CURRENCY\_CODE), the sum of the entered debit amounts (ENTERED\_DR) must equal the sum of the entered credit amounts (ENTERED\_CR).

The Accounting Program balances subledger journal entries by creating an additional line on the entry for each currency requiring balancing if the journal entry is not balanced by entered currency. Such lines use the Entered Currency Balancing Account specified in the ledger definition.

## Balanced by Balancing Segment

When a journal entry has lines whose code combination identifiers (CODE\_COMBINATION\_ID) belong to different balancing segments, the subledger journal entry must be balanced based on the accounted amount for each balancing segment. This means that for each balancing segment, the sum of the accounted debit amounts of the subledger journal entry lines (ACCOUNTED\_DR) must equal the sum

of the accounted credit amounts of the subledger journal entry lines (ACCOUNTED\_CR).

The Accounting Program balances subledger journal entries unbalanced by balancing segment by creating debit and credit balancing segment lines. These lines use accounts defined for the ledger in the Intracompany Balancing Rules page in the Accounting Setup Manager (ASM). The Enable Intracompany Balancing option must be selected in the ledger definition in order to enable the application of the balancing rules. The Accounting Program does not provide an automatic balancing mechanism for manual subledger journal entries unbalanced by the balancing segment. Manual subledger journal entries must be balanced by balancing segment at the time of creation.

See: Intracompany Balancing Rules, *Oracle Financials Implementation Guide*

Intercompany balancing rules are not applied to encumbrance accounting entries.

### Balanced Encumbrance Entries

The Accounting Program balances encumbrance balance type entries using the reserve for encumbrance account. Each encumbrance journal is balanced by balancing segment value (BSV), encumbrance type, and reserve for encumbrance (RFE) account.

**Note:** The reserve for encumbrance account is defined in General Ledger and is only required when budgetary control is enabled.

**Note:** The encumbrance type must be defined and enabled in General Ledger.

If you use normal non-encumbrance accounting for secondary ledger, then attach Non-Encumbrance Subledger Accounting method (Standard Cash or Standard Accrual) to secondary ledger with budgetary flag set to No. If you do not want any accounting for secondary ledger as in PO, then attach a Non-Encumbrance Subledger Accounting method (Standard Cash or Standard Accrual) to secondary ledger with no AAD.

Payables does not provide any AAD for Standard Cash or Standard Accrual.

Accounting ignores secondary ledger and accounts only for primary ledger in this case.

### Validation of Monetary Amounts (ENTERED\_DR, ENTERED\_CR, ACCOUNTED\_DR, ACCOUNTED\_CR)

All subledger journal entries must have valid monetary amounts. A journal entry line has two types of monetary amounts, the entered amount and the accounted amount (also known as the ledger amount).

The accounted amount is always in the currency of the ledger, and the entered amount is always in the currency of the transaction. The transaction currency is stored in each subledger journal entry line.

Both entered and accounted amounts have a side. The side is either a debit or credit. In the Subledger Accounting data model, the side of the amount is not treated as an independent column. The side of the amount is included in the name of the column; entered\_dr and entered\_cr for the entered amount, and accounted\_dr and accounted\_cr for the accounted amount.

The following rules apply to subledger journal entry line monetary amounts:

- They must be signed numbers (including zero).
- They must have the same decimal precision as their associated currency.

For example, if the ledger currency has a decimal precision of two, the monetary amounts must have a decimal precision of two.

- An amount must always be either a debit or a credit. Either the debit or the credit must have a value, including zero.
- Both entered and accounted amounts must be on the same side. In other words, if an entered amount is a debit, then its associated accounted amount must also be a debit and vice versa.

However, the entered and accounted amounts can have different signs. For example, a journal entry line may have a debit rounding difference in the entered amount and a credit rounding difference in the accounted amount. In this case, one single rounding line is created with a positive entered debit amount and a negative accounted debit amount.

- A journal entry whose entered currency is the same as the ledger currency should have the same entered and accounted amounts.

**Note:** This rule does not apply to a rounding line.

- Validation of journals created via the accounting engine in budgetary control mode ensures that budgetary control is executed. Budgetary control validation includes Projects, Grants, Contract Commitments, and General Ledger budgetary control when required.

## Processing Funds Validation Results

When journal validation is completed the accounting engine processes any budgetary control failures or warnings. Failures occur at line level and depending on the mode of budgetary control, the entire journal can be marked as Failed or individual lines can be marked as Failed and the header is marked with a warning. The table below describes the validation results messages.

### **Validation Result Messages**

| <b>Message</b>                              | <b>Description</b>   |
|---|--|
| Successful Budgetary Control - PASS         | All journal lines pass budgetary control and this should be reflected in a message on the journal header. The journal is valid.  |
| Failed Budgetary Control - FAIL             | One or more lines in the journal failed and budgetary control options indicate the whole transaction should fail. The journal is considered in error and this is reflected in a message on the header of the journal. A message is displayed for each line that fails budgetary control. The journal has errors and is therefore not valid.  |
| Advisory Budgetary Control - ADVISORY FAIL  | One or more lines in the journal failed and budgetary control options indicate an advisory message should be displayed but the journal is still valid. A message for the header of the journal entry indicates an advisory failure and an advisory budgetary control message is displayed for each line that failed.   |
| Partial Failure Budgetary Control - PARTIAL | One or more lines in the journal failed and budgetary control options indicate individual lines should be processed separately and part of the transaction should pass. For the same transaction, this results in some lines passing budgetary control while others fail. The journal is valid. A warning message is shown on the header of the journal entry. The specific lines that failed contain a message indicating that they failed budgetary control and the amount for the line is set to zero. For encumbrance journals, the reserve for encumbrance account also needs to be adjusted to balance the journal against the failed lines reset to zero. |
| Budgetary Control Fatal/Unexpected Error    | A system error was encountered during the budgetary control validation routine. The error may be related to the database, middle tier, or other underlying infrastructure errors.  |

| Message                       | Description   |
|-------------------------------|---|
| No Budgetary Control Required | Budgetary control is enabled at ledger level but budgetary control options are set at the account code combination, Project, Grant, or Contract Commitment level. Budgetary control validation is automatically initiated in these cases. |

## Example - Valid Debits and Credits

### 1. Entered currency and ledger currency are the same

The following table lists the details of revenue and receivable lines of a subledger journal entry where the entered currency is the same as the ledger currency.

| Account       | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (USD) | Accounted Cr (USD) | Unrounded Entered DR | Unrounded Entered CR | Unrounded Accounted DR | Unrounded Accounted CR |
|---------------|------------------|------------------|--------------------|--------------------|----------------------|----------------------|------------------------|------------------------|
| 01-Receivable | 100.00           | Null             | 100.00             | Null               | 100.00               | Null                 | 100.00                 | Null                   |
| 01-Revenue    | Null             | 0                | Null               | 0                  | Null                 | 0                    | Null                   | 0                      |
| 01-Revenue1   | Null             | 33.33            | Null               | 33.33              | Null                 | 33.33333             | Null                   | 33.33333               |
| 01-Rounding   | Null             | 0.01             | Null               | 0.01               | Null                 | Null                 | Null                   | Null                   |

### 2. Entered currency is different from the ledger currency

The following table lists the details of revenue and receivable lines of a subledger journal entry where the entered and ledger currencies are different. A conversion rate of 1.1 is assumed.

| Account               | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (EUR) | Accounted Cr (EUR) | Unrounded Entered DR | Unrounded Entered CR | Unrounded Accounted DR (EUR) | Unrounded Accounted CR (EUR) |
|-----------------------|------------------|------------------|--------------------|--------------------|----------------------|----------------------|------------------------------|------------------------------|
| 01-Receivable         | 70.09            | Null             | 56.77              | Null               | 70.088               | Null                 | 56.77128                     | Null                         |
| 01-Revenue            | Null             | 0                | Null               | 0                  | Null                 | 0                    | Null                         | 0                            |
| 01-Revenue2           | Null             | 35.04            | Null               | 28.39              | Null                 | 35.044               | Null                         | 28.38564                     |
| 01-Rounding           | <0.01>           | Null             | 0.01               | Null               | Null                 | Null                 | Null                         | Null                         |
| 01-Exchange Gain/Loss | 0                | Null             | 28.39              | Null               | 0                    | Null                 | 28.38564                     | Null                         |

## Example - Invalid Debits and Credits

The following are examples of incorrect journal line debits and credits. These lines should never be created in Subledger Accounting.

**Entered currency and ledger currency are the same**

| Account    | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (USD) | Accounted Cr (USD) | Reason   |
|------------|------------------|------------------|--------------------|--------------------|--|
| 01-Revenue | 100              | 200              | 100                | 200                | The debit and credit cannot be populated at the same time for the same amount. |

| Account    | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (USD) | Accounted Cr (USD) | Reason  |
|------------|------------------|------------------|--------------------|--------------------|---|
| 01-Revenue | 0                | 0                | 100                | Null               | The entered amount is populated on both debit and credit sides.                               |
| 01-Revenue | Null             | Null             | Null               | Null               | The amounts cannot be null simultaneously.  |
| 01-Revenue | 100              | Null             | Null               | 100                | The entered and accounted amounts must be on the same side (either debit or credit).          |
| 01-Revenue | 100              | Null             | Null               | -100               | The entered and accounted amounts must be on the same side (either debit or credit).          |
| 01-Revenue | 0                | Null             | Null               | Null               | An accounted amount must be populated for every entered amount, even when the amount is zero. |
| 01-Revenue | Null             | Null             | Null               | 0                  | The entered amount must be populated.   |

| Account    | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (USD) | Accounted Cr (USD) | Reason  |
|------------|------------------|------------------|--------------------|--------------------|---|
| 01-Revenue | 0                | Null             | 100                | Null               | If the entered currency is same as the ledger currency, then entered and accounted amounts cannot be different. |

**Entered currency is different from the ledger currency**

| Account    | Entered Dr (USD) | Entered Cr (USD) | Accounted Dr (EUR) | Accounted Cr (EUR) | Reason  |
|------------|------------------|------------------|--------------------|--------------------|---|
| 01-Revenue | Null             | Null             | 100                | Null               | Both entered and accounted amounts must be populated. |

---

# Index

## A

---

Account Analysis Reports, 9-20  
account derivation rules, 2-57  
    Accounting Flexfield rules, 2-58  
    conditions, 2-70  
        examples, 2-71  
        lookup types, 2-71  
    define, 2-64  
    examples, 2-61  
        Accounting Flexfield rules, 2-61  
        segment rules, 2-62  
    segment rules, 2-59  
    value set rules, 2-60  
Account Derivation Rules window, 2-65  
accounting attributes guideline, 2-14  
accounting chart of accounts, 2-8  
    assigning, 2-11  
    mapping, 2-11  
    using, 2-10  
Accounting Class Assignments window, 2-16  
accounting definitions inquiry, 2-121  
    features, 2-122  
accounting events inquiry  
    process steps, 7-2  
Accounting Flexfield Rules  
    account derivation rules, 2-58  
Accounting Methods Builder  
    *See* AMB  
Accounting Methods Builder Components  
diagram, 2-3  
accrual reversal GL date, 2-99

accrual reversals, 2-99  
Complete Multiperiod Accounting process, 2-106  
examples, 2-104  
process, 2-99  
amb  
    multiperiod accounting, 2-90  
AMB  
    accounting chart of accounts, 2-8  
    Accounting Methods Builder Components  
    diagram, 2-3  
    application accounting definitions, 2-107  
    application accounting definitions (AAD)  
    loader import overview, 2-114  
    assigning the transaction and accounting chart  
    of accounts, 2-11  
    business flows, 2-17  
    chart of accounts mapping, 2-11  
    charts of accounts, 2-7  
    Components for Building Accounting  
    Definitions diagram, 2-4  
    copy and modify functionality, 2-6  
    custom sources, 2-12  
    Custom Sources window, 2-13  
    defining journal line types, 2-31  
    Elements of a Subledger Journal Entry  
    diagram, 2-2  
    journal entry descriptions, 2-47  
    journal lines definitions, 2-79  
    journal line types, 2-29  
    journal line types conditions, 2-41  
    mapping sets, 2-51  
    Multiple Accounting Representations

diagram, 2-9  
multiple representations, 2-8  
    example, 2-9  
process, 2-4  
    steps, 2-5  
subledger accounting methods, 2-118  
transaction chart of accounts, 2-7  
using the transaction and accounting chart of accounts, 2-10  
application accounting definitions, 2-107  
    Application Accounting Definitions diagram, 2-108  
    Application Accounting Definitions window, 2-109  
    copy, 2-113  
    copy and modify, 2-112  
    copy to, 2-113  
    Copy to window, 2-113  
    define  
        prerequisite, 2-109  
    modify, 2-113  
    validate, 2-113  
application accounting definitions loader  
    installation, patching, and migration  
        export, 2-116  
applications accounting definitions  
    define, 2-109

## B

---

Basic Business Flow Example diagram, 2-18  
business flows, 2-17  
    Basic Business Flow Example diagram, 2-18  
    business flow class, 2-20  
    business flow method, 2-19  
        prior entry, 2-19  
        same entry, 2-19  
    copy from prior entry  
        example, 2-22  
    copy from prior journal entry, 2-22  
    copy from same entry, 2-26  
        example, 2-26  
    features, 2-19  
Journal Lines Accounting Attributes window, 2-21  
Journal Lines Definitions window, 2-21  
Journal Line Types window, 2-21

setup considerations and tasks, 2-21  
setup features, 2-20

## C

---

chart of accounts mapping, 2-11  
charts of accounts, 2-7  
    accounting, 2-8  
    transaction, 2-7  
Complete Multiperiod Accounting process, 2-106  
Components for Building Accounting Definitions diagram, 2-4  
copy and modify functionality, 2-6  
Copy to window, 2-113  
Create Accounting program  
    Oracle Subledger Accounting Program Report, 3-5  
        overview, 3-1  
        parameters, 3-1  
    custom sources, 2-12  
        define, 2-13  
Custom Sources window, 2-13

## D

---

diagnostic framework  
    Diagnostic Framework Business Process Flow diagram, 5-1  
        features, 5-3  
        execution, 5-3  
        purge, 5-5  
        SLA: Enable Diagnostics, 5-3  
    introduction, 5-1  
    Transaction Objects Diagnostics Report, 5-3  
diagrams, 11-2  
    Accounting Events Inquiry Process Flow, 7-1  
    Accounting Methods Builder Components, 2-3  
    Application Accounting Definitions, 2-108  
    Basic Business Flow Example, 2-18  
    Components for Building Accounting Definitions, 2-4  
    Diagnostic Framework Business Process Flow, 5-1  
        Elements of a Subledger Journal Entry, 2-2  
        Journal Entry Headers Process Flow, 7-4  
        Journal Entry Lines Inquiry Process Flow, 7-5  
        Multiperiod Accounting Process Flow, 2-91  
    Multiple Accounting Representations, 2-9

Reporting Sequence Process Flow, 8-3  
drilldown from GL, 7-7

## **E**

---

Elements of a Subledger Journal Entry diagram, 2-2  
events  
    introduction, 2-12  
export AADs, 2-116

## **I**

---

inquiries  
    Accounting Events Inquiry Process Flow diagram, 7-1  
    headers inquiry  
        process steps, 7-4  
    Journal Entry Headers Process Flow Diagram, 7-4  
    Journal Entry Lines Inquiry Process Flow diagram, 7-5  
    journal entry lines inquiry process steps, 7-6  
    overview, 7-1  
subledger journal entries  
    headers, 7-4  
    lines, 7-5  
    Subledger Journal Entry Lines window, 7-6  
inquiry  
    accounting events  
        process steps, 7-2

## **J**

---

Journal Entries Report, 9-3  
    data elements, 9-11  
    parameters, 9-4  
journal entry description  
    conditions, 2-51  
    copy, 2-51  
    define, 2-48  
    Journal Entry Description window, 2-48  
journal entry description details  
    define, 2-49  
    Journal Entry Description Details window, 2-49  
journal entry descriptions, 2-47  
Journal Entry Headers Process Flow diagram, 7-4

Journal Entry Lines Inquiry Process Flow diagram, 7-5  
Journal Line Accounting Attribute Assignments, 2-31  
journal lines definitions, 2-79  
    Copy Journal Lines Definition window, 2-88  
    define, 2-82  
        prerequisites, 2-81  
        examples, 2-79  
    Journal Lines Definitions window, 2-82  
journal line types, 2-29  
    conditions, 2-41  
        define, 2-43  
        examples, 2-41  
    copy, 2-47  
    define, 2-31  
    features, 2-29  
    Journal Line Accounting Attribute Assignments, 2-31  
    Journal Line Types window, 2-31  
    Journal Line Types window, 2-31

## **L**

---

Line Assignments window, 2-95

## **M**

---

mapping sets, 2-51  
    define, 2-53  
    examples, 2-55  
    Mapping Sets window, 2-53  
    prerequisite, 2-52  
multiperiod accounting, 2-90  
    Complete Multiperiod Accounting process, 2-106  
    define, 2-95  
    define line assignments, 2-95  
    example, 2-93  
    Line Assignments window, 2-95  
    Multiperiod Accounting Process Flow diagram, 2-91  
    Multiperiod Accounting window, 2-95  
    process, 2-90  
    process steps, 2-91  
Multiperiod Accounting and Accrual Reversal Report, 9-47  
Multiperiod Accounting Report (Detail), 9-47

Multiperiod Accounting Report (Summary), 9-47  
Multiperiod Accounting window, 2-95  
Multiple Accounting Representations diagram, 2-9  
multiple representations, 2-8  
example, 2-9

## N

---

navigation paths, A-1

## O

---

Open Account Balances  
open account balances data manager, 10-5  
procedure, 10-5  
Open Account Balances Listing, 10-5  
create definitions, 10-2  
data elements, 10-8  
introduction, 10-1  
parameters, 10-6  
process steps, 10-1  
Oracle Subledger Accounting Program Report, 3-5

## P

---

Post-Accounting Programs, 2-15  
Post-Accounting Programs window, 2-16  
procedures  
copy, modify, and validate application accounting definitions, 2-113  
copy a subledger accounting method, 2-121  
copy journal line types and their associated conditions, 2-47  
create or update Open Account Balances  
Listing definitions, 10-3  
create subledger journal entries, 6-2  
create transaction account definitions, 11-4  
define account derivation rules for Accounting Flexfields, 2-65  
define a journal entry description, 2-48  
define custom sources, 2-13  
define journal entry description details, 2-49  
define journal lines definitions, 2-82  
define journal line type conditions, 2-43  
define mapping sets, 2-53  
define Post-Accounting Programs, 2-15

define subledger accounting method, 2-119  
run the Open Account Balances Data Manager program, 10-5  
to define journal line types, 2-31  
update Subledger Accounting options, 1-3  
update system options, 1-3  
profile option categories  
overview, B-1  
profile options  
descriptions, B-2  
overview, B-1  
SLA: Accounting Methods Builder Context, B-3  
SLA: Additional Data Access, B-8  
SLA: Disable Journal Import, B-7  
SLA: Display Timestamp on Trace, B-5  
SLA: Enable Data Access Security in Subledgers, B-8  
SLA: Enable Diagnostics, B-5  
SLA: Enable PL/SQL Profiler, B-5  
SLA: Enable SQL Trace, B-5  
SLA: Enable SRS Log/Output, B-5  
SLA: Enable Subledger Transaction Security in GL, B-9  
SLA: Enable Trace, B-5  
SLA: Flush Trace File, B-6  
SLA: Initial Date for Historical Upgrade, B-7  
SLA: Oracle Forms Trace Mode, B-6  
SLA: SRS Trace Mode, B-6  
SLA: Trace File Override Directory, B-6  
SLA: Trace File Override Filename, B-6  
SLA: Trace Level, B-6  
SLA: Trace Timeout, B-6

## R

---

reporting sequence  
overview, 8-1  
process steps, 8-4  
Reporting Sequence Process Flow diagram, 8-3  
sequence programs, 8-2  
reports  
Account Analysis, 9-20  
Export Application Accounting Definitions, 2-118  
Import Application Accounting Definitions, 2-

116  
Journal Entries Report, 9-3  
Multiperiod Accounting and Accrual Reversal Report, 9-47  
Multiperiod Accounting Report (Detail), 9-47  
Multiperiod Accounting Report (Summary), 9-47  
Open Account Balances Listing, 10-1, 10-5  
Oracle Subledger Accounting Program Report, 3-5  
Subledger Accounting Rules Detail Listing, 9-66  
Subledger Period Close Exceptions, 9-62  
Third Party Balance, 9-33  
Transaction Objects Diagnostics, 5-3

## S

---

seed data  
Accounting Class Assignments window, 2-16  
Post-Accounting Programs, 2-15  
downloading, 2-17  
Post-Accounting Programs window, 2-16  
segment rules  
account derivation rules, 2-59  
sequence programs, 8-2  
SLA: Accounting Methods Builder Context, B-3  
SLA: Additional Data Access Set, B-8  
SLA: Disable Journal Import, B-7  
SLA: Display Timestamp on Trace, B-5  
SLA: Enable Data Access Security in Subledgers, B-8  
SLA: Enable Diagnostics, B-5  
SLA: Enable SQL Trace, B-5  
SLA: Enable SRS Log/Output, B-5  
SLA: Enable Subledger Transaction Security in GL, B-9  
SLA: Enable Trace, B-5  
SLA: Flush Trace File, B-6  
SLA: Initial Date for Historical Upgrade, B-7  
SLA: Oracle Forms Trace Mode, B-6  
SLA: SRS Trace Mode, B-6  
SLA: Trace File Override Directory, B-6  
SLA: Trace File Override Filename, B-6  
SLA: Trace Level, B-6  
SLA: Trace Timeout, B-6  
subledger accounting method  
copy and modify, 2-120  
Copy to window, 2-121  
define, 2-119  
Subledger Accounting Method window, 2-119  
subledger accounting methods, 2-118  
Subledger Accounting options  
option descriptions, 1-3  
prerequisites, 1-2  
setup pages, 1-1  
system options, 1-1  
to update Subledger Accounting options, 1-3  
to update system options, 1-3  
Update Accounting Options window, 1-3  
Subledger Accounting options setup  
overview, 1-1  
subledger accounting reports  
introduction, 9-1  
Subledger Accounting Rules Detail Listing  
parameters, 9-67  
Subledger Accounting Rules Detail Listing Report  
data elements, 9-67  
Subledger Period Close Exceptions, 9-62  
parameters, 9-63  
Subledger Period Close Exceptions Report  
data elements, 9-64  
XML output file, 9-2  
Subledger Accounting reports  
Account Analysis, 9-20  
data elements, 9-24  
parameters, 9-21  
Journal Entries, 9-3  
Journal Entries Report  
data elements, 9-11  
parameters, 9-4  
Multiperiod Accounting, 9-47  
Third Party Balances Report, 9-33  
Subledger Accounting Rules Detail Listing Report, 9-66  
subledger accounting rules detail report  
Subledger Accounting Rules Detail Listing Report, 9-66  
subledger journal entries  
create, 6-2  
headers inquiry, 7-4  
lines inquiry, 7-5  
overview, 6-1

Subledger Journal Entry Lines window, 7-6  
viewing supporting reference balances, 7-7  
subledger journal entry  
definition, C-1  
balancing rules, C-14  
header validation, C-2  
high level, C-1  
line validation, C-9  
monetary amounts validation, C-16  
Subledger Journal Entry Lines window, 7-6  
Subledger Period Close Exceptions Report, 9-62  
Supporting References, 2-72  
defining, 2-76  
deleting, 2-78  
duplicating, 2-78  
Example, 2-73  
Setup, 2-73  
updating, 2-77  
Viewing Supporting Reference Balances, 7-7  
Year-End Carry Forward, 2-76  
system options, 1-1

## T

---

TAB  
introduction, 11-1  
setup, 11-3  
Transaction Account Builder Components  
diagram, 11-2  
transaction account definitions, 11-1  
create, 11-4  
Third Party Balances Report, 9-33  
transaction account builder  
*See TAB*  
Transaction Account Builder Components, 11-2  
Transaction Account Builder Components  
diagram, 11-2  
transaction account definitions, 11-1  
create, 11-4  
transaction chart of accounts, 2-7  
assigning, 2-11  
mapping, 2-11  
using, 2-10  
Transaction Objects Diagnostics Report, 5-3  
transaction objects guidelines  
accounting attributes, 2-14  
Transfer Journal Entries to General Ledger

program  
overview, 3-1  
Transfer Journal Entries to GL Program, 3-4  
Transfer Journal Entries to GL Report, 3-5

## U

---

Update Accounting Options window, 1-3

## V

---

value set rules  
account derivation rules, 2-60

## W

---

windows  
Account Derivation Rules, 2-65  
Accounting Class Assignments, 2-16  
Application Accounting Definitions, 2-109  
Copy to, 2-113  
subledger accounting method, 2-121  
Custom Sources, 2-13  
Journal Entry Description, 2-48  
Journal Entry Description Details, 2-49  
Journal Line Accounting Attribute  
Assignments, 2-31  
Journal Lines Definitions, 2-82  
Journal Line Types, 2-31  
Line Assignments, 2-95  
Mapping Sets, 2-53  
Multiperiod Accounting, 2-95  
Post-Accounting Programs, 2-16  
Subledger Accounting Method, 2-119  
Subledger Journal Entry Lines, 7-6  
Update Accounting Options, 1-3

## X

---

XML output file, 9-2