

Oracle® Application Integration Architecture

Oracle Financial Operations Control Integration Pack
Implementation Guide for Oracle Retail Merchandise Operations
Management and PeopleSoft Enterprise Financials

Release 3.1

E20500-02

January 2012

Oracle Application Integration Architecture Oracle Financial Operations Control Integration Pack
Implementation Guide for Oracle Retail Merchandise Operations Management and PeopleSoft Enterprise
Financials, Release 3.1

E20500-02

Copyright © 2001, 2012, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

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 RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

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

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

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

Preface	xiii
Common Oracle AIA Pre-Built Integration Guides	xiii
Documentation Accessibility	xiii
Additional Resources	xiv
Part I Understanding the Delivered Integrations	
1 Oracle Financial Operations Control Integration Pack Overview	
1.1 Understanding the Oracle Financial Operations Control Integration Pack	1-1
1.2 Key Benefits	1-3
1.3 Participating Applications	1-4
1.3.1 Oracle RMS Overview	1-4
1.3.2 Oracle ReSA Overview	1-4
1.3.3 Oracle ReIM Overview	1-5
1.3.4 PeopleSoft Payables Overview	1-5
1.3.5 PeopleSoft GL Overview	1-5
1.4 Retail Sales Financial Business Process Flow	1-5
1.5 Retail Inventory Financial Business Process Flow	1-6
1.6 Retail Merchandise Procure to Pay Business Process Flow	1-8
1.7 Solution Assumptions and Constraints	1-10
2 Life Cycle Management for Reference Data	
2.1 Currency Exchange Rate Integration	2-1
2.1.1 Currency Exchange Rate Integration Overview	2-1
2.1.1.1 Solution Assumptions and Constraints	2-2
2.1.2 Currency Exchange Rate Integration Details	2-3
2.1.2.1 Sequence Diagram	2-3
2.1.3 PeopleSoft GL Interfaces	2-5
2.1.4 Oracle RIB Interfaces	2-5
2.1.5 Core AIA Components	2-5
2.1.6 Integration Services	2-6
2.1.6.1 CurrencyExchangePeopleSoftJMSConsumer	2-6
2.1.6.2 SyncCurrencyExchangeListPeopleSoftReqABCImpl	2-7
2.1.6.3 CurrencyExchangeEBS	2-8
2.1.6.4 SyncCurrencyExchangeListRetailProvABCImpl	2-9

2.1.6.5	SyncCurrencyExchangeListRetailProvJMSProducer	2-10
2.2	Payment Term Integration.....	2-11
2.2.1	Payment Term Integration Overview	2-11
2.2.1.1	Solution Assumptions and Constraints.....	2-12
2.2.2	Payment Term Integration Details	2-13
2.2.2.1	Sequence Diagram	2-13
2.2.2.2	Multiple Language Support.....	2-15
2.2.3	PeopleSoft Payables Interfaces.....	2-16
2.2.4	Oracle RMS Interfaces	2-16
2.2.5	Core AIA Components	2-17
2.2.6	Integration Services	2-17
2.2.6.1	PaymentTermPeopleSoftJMSPConsumer	2-18
2.2.6.2	SyncPaymentTermListPeopleSoftReqABCImpl	2-18
2.2.6.3	CreatePaymentTermPeopleSoftReqABCImpl.....	2-19
2.2.6.4	UpdatePaymentTermPeopleSoftReqABCImpl	2-19
2.2.6.5	PaymentTermEBS.....	2-20
2.2.6.6	SyncPaymentTermListRetailProvABCImpl	2-20
2.2.6.7	CreatePaymentTermRetailProvABCImpl.....	2-21
2.2.6.8	UpdatePaymentTermRetailProvABCImpl	2-22
2.2.6.9	PayTermService	2-22
2.3	Suppliers Information Integration.....	2-23
2.3.1	Supplier Integration Overview	2-23
2.3.1.1	Solution Assumptions and Constraints.....	2-24
2.3.2	Supplier Integration Details	2-25
2.3.2.1	Sequence Diagram	2-26
2.3.2.2	Multiple Language Support.....	2-27
2.3.3	PeopleSoft Payables Interfaces.....	2-28
2.3.4	Oracle RMS Interfaces	2-28
2.3.5	Core AIA Components	2-28
2.3.6	Integration Services	2-29
2.3.6.1	SupplierPartyPeopleSoftJMSPConsumer	2-29
2.3.6.2	SyncSupplierPartyListPeopleSoftReqABCImpl.....	2-30
2.3.6.3	CreateSupplierPartyPeopleSoftReqABCImpl	2-30
2.3.6.4	UpdateSupplierPartyPeopleSoftReqABCImpl.....	2-31
2.3.6.5	SupplierPartyEBS	2-32
2.3.6.6	SyncSupplierPartyListRetailProvABCImpl.....	2-32
2.3.6.7	CreateSupplierPartyRetailProvABCImpl	2-33
2.3.6.8	UpdateSupplierPartyRetailProvABCImpl.....	2-34

3 Chart of Accounts Validation Integration Flow

3.1	Requesting Chart of Accounts Combination Validation.....	3-1
3.1.1	Solution Assumptions and Constraints.....	3-2
3.1.2	Chart of Accounts Combination Validation Integration Flow.....	3-2
3.2	Oracle Retail Interfaces	3-4
3.3	PeopleSoft GL Interfaces	3-4
3.4	Core AIA Components.....	3-4
3.5	Integration Services	3-5

3.5.1	ProcessGLAccountValidationRetailReqABCImpl.....	3-5
3.5.2	ChartOfAccountsEBS	3-6
3.5.3	ProcessGLAccountValidationPeopleSoftProvABCImpl.....	3-6

4 Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries

4.1	Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview	4-1
4.1.1	Business Process Flow for the Integration of Accounting Entries	4-2
4.1.2	Facts and Constraints	4-2
4.2	Performing Setup Tasks	4-3
4.3	Configuring and Generating Data in Oracle Retail	4-3
4.3.1	Configuring the Data in Oracle Retail	4-3
4.3.2	Generating the Data in Oracle Retail	4-4
4.4	Picking Up and Transforming the Data.....	4-4
4.5	Configuring and Running the Process Integration for Accounting Entries.....	4-6
4.5.1	Configuring the Process Integration for Accounting Entries.....	4-6
4.5.1.1	SettingUpAIAConfigurationProperties.XML.....	4-6
4.5.1.2	Setting Up and Exporting DVMs	4-7
4.5.1.3	Creating Views in Retail Database.....	4-7
4.5.2	Setting Up a Schedule	4-8
4.5.3	Running the Process Integration for Accounting Entries	4-9

5 Process Integration for Retail Merchandise Procure to Pay

5.1	Process Integration for Retail Merchandise Procure to Pay Overview	5-1
5.1.1	Business Process Flow for Integration of Retail Merchandise Procure to Pay	5-2
5.1.2	Solution Assumptions and Constraints.....	5-3
5.2	Performing Setup Tasks	5-4
5.2.1	Setup Tasks Specific to Oracle ReIM.....	5-4
5.2.2	Setup Tasks Specific to PeopleSoft Payables	5-4
5.2.3	Setup Tasks Specific to the Integration Layer.....	5-4
5.3	Configuring and Generating Data in Oracle Retail	5-5
5.3.1	Configuring the Data in Oracle Retail	5-5
5.3.2	Generating the Data in Oracle Retail	5-6
5.4	Picking Up and Transforming the Data.....	5-6
5.5	Configuring and Running the Process Integration for Retail Merchandise Procure to Pay....	5-8
5.5.1	Configuring the Process Integration for Retail Merchandise Procure to Pay.....	5-8
5.5.1.1	Setting Up AIAConfigurationProperties.XML.....	5-8
5.5.1.2	Setting Up and Exporting DVMs	5-9
5.5.1.3	Creating Views in Retail Database.....	5-9
5.5.2	Setting Up a Schedule	5-10
5.5.3	Running the Process Integration for Retail Merchandise Procure to Pay	5-11

6 Drill Back and Drill Forward

6.1	Drill Back and Drill Forward Overview	6-1
-----	---	-----

6.1.1	Setup Activities	6-2
6.1.2	Solution Assumptions and Constraints.....	6-3
6.2	Drill Back and Drill Forward Details	6-3
6.2.1	Drill Back Details	6-3
6.2.2	Drill Forward Details	6-4
6.2.3	Overall Flow	6-4
6.3	Service Solution Components	6-5
6.3.1	Service Solution Component Details	6-5

Part II Implementing the Delivered Integrations

7 Configuring the Oracle Financial Operations Control Integration Pack

7.1	Prerequisites	7-2
7.1.1	Currency Exchange Rate Prerequisites.....	7-2
7.1.2	Payment Term Prerequisites	7-2
7.1.3	Supplier Information Prerequisites.....	7-2
7.1.4	Inventory Valuation and Revenue Recognition Accounting Entries Prerequisites...	7-2
7.1.5	Retail Merchandise Procure to Pay Prerequisites	7-3
7.1.6	Drill Back and Drill Forward Prerequisites	7-3
7.2	Data Requirements	7-3
7.2.1	Currency Exchange Rate Data Requirements.....	7-3
7.2.2	Payment Term Data Requirements	7-3
7.2.2.1	Class Diagram	7-4
7.2.3	Supplier Information Data Requirements.....	7-6
7.2.3.1	Class Diagram	7-6
7.2.4	Chart of Accounts Validation Data Requirements	7-9
7.2.5	Inventory Valuation and Revenue Recognition Accounting Entries Data Requirements 7-9	
7.2.6	Retail Merchandise Procure to Pay Data Requirements	7-9
7.2.7	Drill Back and Drill Forward Data Requirements	7-9
7.3	Setting Up Oracle Retail Applications	7-10
7.4	Setting Up PeopleSoft GL	7-10
7.5	Setting Up PeopleSoft Payables	7-10
7.6	Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities	7-10
7.6.1	Populating Cross-References	7-11
7.6.2	Validating Cross-References	7-12
7.6.3	Viewing Cross-Reference Data	7-12
7.7	Setting Up PeopleSoft Domain Value Maps	7-13
7.8	Manually Synchronizing Freight Terms.....	7-13
7.9	Manually Synchronizing Currency Exchange Rate Types	7-14
7.10	Manually Synchronizing Currency Codes	7-14
7.11	Manually Synchronizing Chart of Accounts Values	7-14
7.12	Manually Synchronizing Financial Calendars.....	7-15
7.13	Mapping Business and Organization Units	7-15
7.14	Working with Cross-References	7-15
7.15	Working with DVMs	7-17
7.16	Handling Errors	7-18

7.17	Viewing EIMs	7-18
7.18	Setting Configuration Properties	7-18

List of Tables

2-1	Operations Used by SyncPaymentTermListRetailProvABCServiceImpl	2-21
4-1	Integration of Accounting Entries Overall Process.....	4-2
5-1	Retail Merchandise Procure to Pay Overall Process.....	5-2
6-1	Service Solution Components	6-5
7-1	PeopleSoft DVMs	7-13
7-2	Oracle Retail to PeopleSoft Financials PIP Cross-References.....	7-16
7-3	DVMs for the Oracle Retail to PeopleSoft Financials PIP	7-17
7-4	Errors Generated by the Oracle Retail to PeopleSoft Financials PIP.....	7-18
7-5	ProcessGLAccountValidationPeopleSoftProvABCServiceImpl Service Property.....	7-19
7-6	SyncCurrencyExchangeListPeopleSoftReqABCServiceImpl Service Property.....	7-19
7-7	SyncPaymentTermListPeopleSoftReqABCServiceImpl Service Property	7-20
7-8	CreatePaymentTermPeopleSoftReqABCServiceImpl Service Property.....	7-21
7-9	UpdatePaymentTermPeopleSoftReqABCServiceImpl Service Property	7-21
7-10	CreateSupplierPartyPeopleSoftReqABCServiceImpl Service Property	7-22
7-11	UpdateSupplierPartyPeopleSoftReqABCServiceImpl Service Property.....	7-22
7-12	SyncCurrencyExchangeListRetailProvABCServiceImpl Service Property.....	7-23
7-13	SyncPaymentTermListRetailProvABCServiceImpl Service Property	7-24
7-14	UpdatePaymentTermRetailProvABCServiceImpl Service Property	7-24
7-15	CreatePaymentTermRetailProvABCServiceImpl Service Property.....	7-25
7-16	SyncSupplierPartyListRetailProvABCServiceImpl Service Property.....	7-25
7-17	CreateSupplierPartyRetailProvABCServiceImpl Service Property	7-26
7-18	SyncSupplierPartyListRetailProvABCServiceImpl Service Property.....	7-27
7-19	UpdateSupplierPartyRetailProvABCServiceImpl Service Property.....	7-27
7-20	ProcessGLAccountValidationRetailReqABCServiceImpl Service Property.....	7-28
7-21	DrillBackForward Service Property	7-28

List of Figures

1-1	Oracle Retail to PeopleSoft Financials Process Flow	1-3
1-2	Retail Sales Financial Business Process Flow	1-6
1-3	Oracle Retail Inventory Financials business process flow	1-7
1-4	Retail Merchandise Procure to Pay Business Process Flow (1 of 3).....	1-8
1-5	Retail Merchandise Procure to Pay business process flow (2 of 3)	1-9
1-6	Retail Merchandise Procure to Pay business process flow (3 of 3)	1-9
2-1	Currency exchange rate integration flow	2-3
2-2	Currency exchange rate integration sequence diagram	2-4
2-3	CurrencyExchangePeopleSoftJMSCConsumer relationship in the integration flow	2-7
2-4	SyncCurrencyExchangeListPeopleSoftReqABCImpl relationship in the integration flow ... 2-8	
2-5	CurrencyExchangeEBS relationship in the integration flow	2-9
2-6	SyncCurrencyExchangeListRetailProvABCImpl relationship in the integration flow	2-10
2-7	SyncCurrencyExchangeListRetailProvJMSProducer relationship in the integration flow	2-11
2-8	Payment term integration flow	2-13
2-9	Synchronize payment term sequence flow diagram	2-14
2-10	Relationship of PaymentTermPeopleSoftJMSCConsumer to other services	2-18
2-11	Relationship of SyncPaymentTermListPeopleSoftReqABCImpl to other services	2-19
2-12	Relationship of CreatePaymentTermPeopleSoftReqABCImpl to other services	2-19
2-13	Relationship of UpdatePaymentTermPeopleSoftReqABCImpl to other services	2-20
2-14	Relationship of PaymentTermEBS to other services	2-20
2-15	Relationship of SyncPaymentTermListRetailProvABCImpl to other services	2-21
2-16	Relationship of CreatePaymentTermRetailProvABCImpl to other services	2-22
2-17	Relationship of UpdatePaymentTermRetailProvABCImpl to other services	2-22
2-18	Relationship of PayTermService to other services	2-23
2-19	Supplier integration flow	2-25
2-20	Supplier integration sequence diagram	2-26
2-21	Relationship of SupplierPartyPeopleSoftJMSCConsumer to other services	2-30
2-22	Relationship of SyncSupplierPartyListPeopleSoftReqABCImpl to other services	2-30
2-23	Relationship of CreateSupplierPartyPeopleSoftReqABCImpl to other services	2-31
2-24	Relationship of UpdateSupplierPartyPeopleSoftReqABCImpl to other services	2-32
2-25	Relationship of SupplierPartyEBS to other services	2-32
2-26	Relationship of SyncSupplierPartyListRetailProvABCImpl to other services	2-33
2-27	Relationship of CreateSupplierPartyRetailProvABCImpl to other services	2-34
2-28	Relationship of UpdateSupplierPartyRetailProvABCImpl to other services	2-35
3-1	Chart of accounts combination validation sequence diagram	3-3
4-1	Business process flow for the integration of accounting entries	4-2
4-2	Accounting entries load process flowchart	4-5
4-3	Example of the Datatype conversion to VARGRAPHIC setup	4-9
5-1	Business process flow for Retail Merchandise Procure to Pay	5-2
5-2	Retail Merchandise Procure to Pay process flow	5-7
5-3	Example of the setup of the Datatype conversion to VARGRAPHIC	5-11
6-1	Overall flow diagram for drill back and drill forward	6-4
6-2	Solution components for drill back	6-6
6-3	Solution components for drill forward	6-7
7-1	Payment Term Integration class diagram (1 of 3)	7-4
7-2	Payment Term Integration class diagram (2 of 3)	7-5
7-3	Payment Term Integration class diagram (3 of 3)	7-6
7-4	Class diagram (1 of 3)	7-7
7-5	Class diagram (2 of 3)	7-8
7-6	Class diagram (3 of 3)	7-9

List of Examples

5-1	AIAConfigurationProperties.XML Parameters.....	5-5
6-1	AIAConfigurationProperties Parameters.....	6-3
7-1	Sample Organization.xml	7-11

Preface

Welcome to the Oracle Financial Operations Control Integration Pack Implementation Guide for Oracle Retail Merchandise Operations Management and PeopleSoft Enterprise Financials.

Common Oracle AIA Pre-Built Integration Guides

Oracle Application Integration Architecture Pre-Built Integrations 11.1 includes the following guides shared by all products delivered with this release:

- Oracle Application Integration Architecture Installation and Upgrade Guide for Pre-Built Integrations Release 11.1

This guide provides an overview of the installation process, including how to install, configure, and deploy your pre-built integrations. The steps required to upgrade your pre-built integrations to the latest release are also provided.

- Oracle Application Integration Architecture Pre-Built Integrations 11.1: Utilities Guide

This guide describes:

- How to work with and configure Session Pool Manager (SPM), which is a service in the Oracle SOA Suite web server whose primary function is to manage a pool of web server session tokens that can be reused by BPEL flows.
- How to deploy and configure the AIACompositeScheduler. This is a utility component that is used by pre-built integrations to schedule a service-oriented architecture (SOA) composite to be invoked at the specified time interval.

- Oracle Application Integration Architecture Pre-Built Integrations 11.1: Product-to-Guide Index

The product-to-Guide index lists the guides that provide information for each product delivered in this release.

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.

Additional Resources

The following resources are also available:

- **Oracle Application Integration Architecture Foundation Pack:**

Oracle AIA Pre-Built integrations require Foundation Pack 11.1.1.5.0 to be installed. Refer to the Foundation Pack documentation library on OTN to download the Foundation Pack guides at http://download.oracle.com/docs/cd/E21764_01/aia.htm.

- **Oracle Application Integration Architecture: Product-to-Guide Index:**

Oracle Technology Network:
<http://www.oracle.com/technetwork/index.html>

- **Known Issues and Workarounds:**

My Oracle Support: <https://support.oracle.com/>

- **Release Notes:**

Oracle Technology Network:
<http://www.oracle.com/technetwork/index.html>

- **Documentation updates:**

Oracle Technology Network:
<http://www.oracle.com/technetwork/index.html>

Part I

Understanding the Delivered Integrations

Part I includes the following chapters:

- Chapter 1, "Oracle Financial Operations Control Integration Pack Overview"
- Chapter 2, "Life Cycle Management for Reference Data"
- Chapter 3, "Chart of Accounts Validation Integration Flow"
- Chapter 4, "Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries"
- Chapter 5, "Process Integration for Retail Merchandise Procure to Pay"
- Chapter 6, "Drill Back and Drill Forward"

Oracle Financial Operations Control Integration Pack Overview

This chapter provides an overview of the Oracle Financial Operations Control Integration Pack for Oracle Retail Merchandise Operations Management and PeopleSoft Enterprise Financials. It also discusses key benefits and participating applications, and describes the retail sales financial, retail inventory financial, and retail procure to pay business process flows. Also included are solution assumptions and constraints.

This chapter includes the following sections:

- [Section 1.1, "Understanding the Oracle Financial Operations Control Integration Pack"](#)
- [Section 1.2, "Key Benefits"](#)
- [Section 1.3, "Participating Applications"](#)
- [Section 1.4, "Retail Sales Financial Business Process Flow"](#)
- [Section 1.5, "Retail Inventory Financial Business Process Flow"](#)
- [Section 1.6, "Retail Merchandise Procure to Pay Business Process Flow"](#)
- [Section 1.7, "Solution Assumptions and Constraints"](#)

1.1 Understanding the Oracle Financial Operations Control Integration Pack

The Oracle Financial Operations Control Integration Pack for Oracle Retail Merchandise Operations Management and PeopleSoft Enterprise Financials provides integration to a robust enterprise financial system that complements the Oracle Retail Merchandising system in a retail customer environment.

This process integration includes these four processes:

- Life Cycle Data Management

This process enables users of Oracle Retail Merchandise Suite to perform functions with data that is shared with the PeopleSoft applications throughout the life cycle of the data that is created and updated. This process provides data synchronization for the initial load before implementation, and incremental data creation and maintenance after implementation. PeopleSoft Payables is the source of suppliers, payment terms, currency, and freight terms. This process synchronizes suppliers, payment terms, and currency exchange rates from PeopleSoft applications to the Oracle Retail Merchandising System (Oracle RMS).

Freight terms are static in nature and their volume is very low; therefore, they must be manually synchronized between the two systems.

- Inventory Valuation (Retail stock ledger)

This process enables users to post accounting entries generated from transactions that change the value of sellable products at retailers' inventory locations (stores and warehouses) from the Oracle Retail Merchandising stock ledger to PeopleSoft General Ledger (GL). Valuation of sellable inventory in stores and warehouses is based on the processing of transactions for movement, pricing, costing, and sale of the inventory. This valuation is captured and processed in the Oracle Retail stock ledger. These transactions include sales, shipments from warehouse to store, store receipts, store transfers, returns to vendors, price changes, stock counts, and shrinkage due to theft or damage.

- Retail Revenue Recognition

This process enables users to post accounting entries generated from sales and returns transactions from retailer's stores for revenue and cash reconciliation to the appropriate ledgers. In this process, data flows from Oracle Retail Sales Audit (Oracle ReSA) to PeopleSoft GL. This process records the financial effect of sale and return, cash reconciliation, and void transactions from stores.

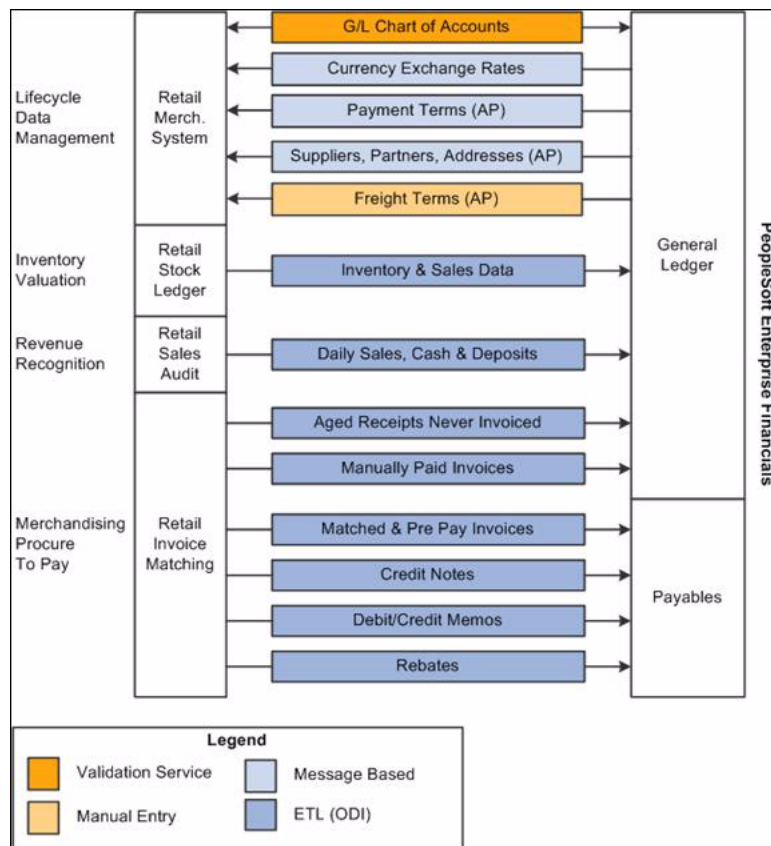
Note: Sales and returns can also come from the Oracle RMS stock ledger.

The Revenue Recognition process begins when Oracle ReSA processes store transactions (sales and returns). For each store transaction, Oracle ReSA generates the appropriate accounting entries to be posted to PeopleSoft GL. Each accounting entry has a valid account code segment combination based on the transaction type, business unit, and location (store or warehouse).

- Retail Merchandising Procure to Pay

This process begins with the Oracle Retail Invoice Matching (Oracle ReIM) application. Invoices from suppliers for retail merchandise are matched to the original purchase order (PO) for the merchandise and the receipt of the merchandise by the retailer. A proper match of invoice, PO, and receipt triggers the payment authorization of the supplier's invoice. Invoices may be authorized for payment before receipt of goods for which prepayment is required. When the authorization for payment is generated, the appropriate accounting distribution is also generated to support the payment authorization. The Retail Merchandise Procure to Pay integration automates the processing of invoice payments, adjustments, and write-offs from Oracle ReIM to PeopleSoft Payables and GL. Other accounting transactions are generated from Oracle ReIM to write off aged receipts that were never invoiced and to post accounting distribution for manually paid or prepaid invoices after receipt.

[Figure 1-1](#) illustrates the Oracle Retail to PeopleSoft Financials process flow.

Figure 1–1 Oracle Retail to PeopleSoft Financials Process Flow

This PIP does not synchronize charts of accounts from PeopleSoft GL to Oracle Retail. Charts of accounts are combinations of account code segments. Because transaction types are defined and assigned combinations of code segments for proper handling of the financial effects in Oracle Retail, PeopleSoft GL provides a service that validates the code combinations. This service ensures that the accounting entries generated by the transactions are valid when they are posted to PeopleSoft GL.

The PIP also provides the drill back drill forward feature. This feature enables the PeopleSoft GL and Oracle Retail users to access information about the financial transactions from the integrated systems in a seamless and intuitive manner. It provides users an end-to-end traceability between PeopleSoft Payables and General Ledger back to Oracle RMS, Oracle ReSA, and Oracle ReIM. From PeopleSoft Enterprise Financials, users can drill back to query data from Oracle Retail applications. The drill back is initiated from the user interfaces (UI) of PeopleSoft Financials. Oracle Retail application users can drill forward to query integrated data from PeopleSoft Enterprise Financials. The drill back and drill forward feature provides traceability from retail transactions to journal entries, and back, and from invoices to vouchers, and back, between Oracle Retail and PeopleSoft Financials. This capability enables processes to support supplier servicing and financial auditing.

1.2 Key Benefits

Here are the key benefits of this PIP:

- This integration is not a point-to-point integration between the PeopleSoft Financials and Oracle Retail applications.
This PIP implementation is independent of the version of integrated applications. An Oracle AIA layer serves as an intermediate thin layer of application between PeopleSoft Financials and Oracle Retail. This integration remains synchronized with the new releases of the edge applications.
- The system exports the sales audit data to the PeopleSoft Financial applications days before the traditional audit process permits.
The Financials applications can use this timely data proactively, which results in increased productivity and operational efficiencies.
- This PIP reduces the total cost of ownership for Oracle and its customers.
- Drill back and drill forward actions between the PeopleSoft Financials and Oracle Retail applications are seamless.
The Oracle Retail and PeopleSoft Financials users do not require any training or must inquire about specific data in either application.
- This PIP supports cost or retail accounting methods.

1.3 Participating Applications

This section discusses these topics:

- [Section 1.3.1, "Oracle RMS Overview"](#)
- [Section 1.3.2, "Oracle ReSA Overview"](#)
- [Section 1.3.3, "Oracle ReIM Overview"](#)
- [Section 1.3.4, "PeopleSoft Payables Overview"](#)
- [Section 1.3.5, "PeopleSoft GL Overview"](#)

1.3.1 Oracle RMS Overview

Oracle RMS is an integrated solution for global retailing. This solution enables retailers to better manage, control, and perform crucial day-to-day merchandising activities. From new product introduction to inventory management, Oracle RMS provides retailers with a complete end-to-end solution and is the most comprehensive and integrated solution for global retailing.

For more information about Oracle RMS, see the *OracleRetail Merchandising System User Guide*.

1.3.2 Oracle ReSA Overview

Oracle ReSA provides retailers with a flexible tool that evaluates and ensures accuracy and completeness of point of sale (POS) data. Realtime access to this audited sales data ensures integrity of information throughout the retail enterprise. With a highly configurable sales audit application, the retailer can maintain existing business practices while providing for future options as the operations grow and change.

ReSA enables retailers to receive POS transaction data, cleanse it, and export the data to the Oracle Merchandising system and the Oracle Retail Data Warehouse. By providing corporate control and visibility to sales audit information, Oracle ReSA enables retailers to make better decisions to improve merchandise operations and transform the economics of their business.

For more information about Oracle ReSA, see the *Oracle Retail Sales Audit User Guide*.

1.3.3 Oracle ReIM Overview

Oracle ReIM is a market-leading solution for retailers who require an automated application to better manage reconciliation and payment of invoices. This advanced solution enables account payables teams to resolve discrepancies on invoices quickly before payments are made. A highly automated, multidimensional matching engine minimizes time spent on manual reviews. Automated routing provides an effective method to ensure that accurate information is delivered to the correct internal teams for resolution and compliance controls.

For more information about Oracle ReIM, see the *Oracle Retail Invoice Matching User Guide*.

1.3.4 PeopleSoft Payables Overview

PeopleSoft Payables provides automated invoice and payment processing to ensure timely and accurate payment for goods and services. Best-practice business processes match purchase orders, receipts, and invoices and provide online approvals to identify exceptions and increase control over disbursements.

PeopleSoft Payables delivers built-in controls to help an enterprise meet regulatory requirements, enforce compliance, reduce risk, and implement due-diligence best practices, thereby reducing cycle times and errors. Other features include a flexible, user-defined system setup; extensive vendor maintenance; digital signatures and financials sanction validation; and powerful inquiry and analytical capabilities.

For more information about PeopleSoft Payables, see the *PeopleSoft Enterprise Payables PeopleBook*.

1.3.5 PeopleSoft GL Overview

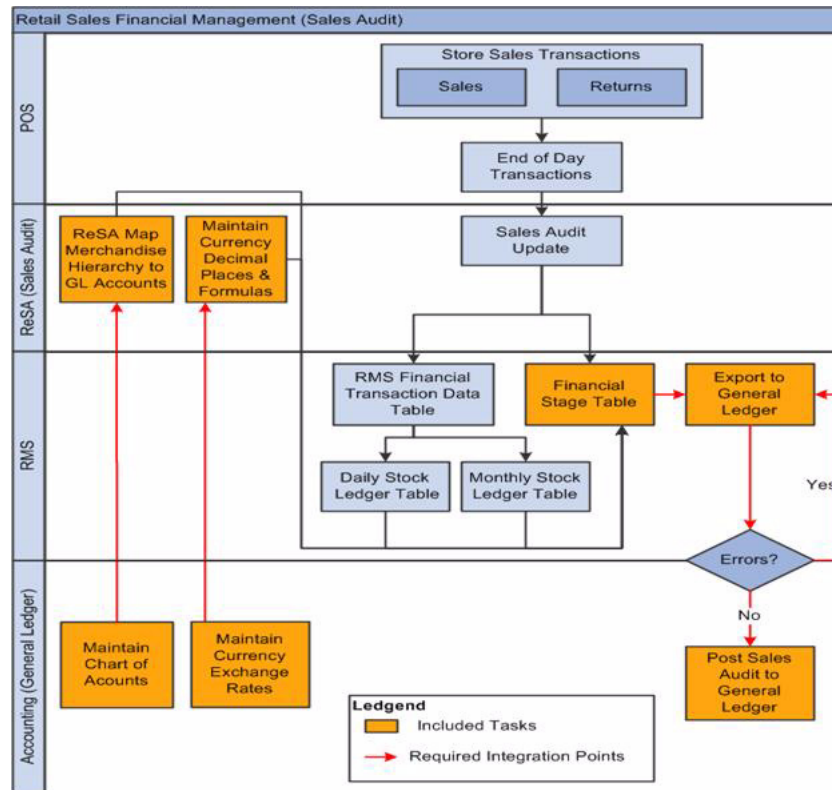
PeopleSoft GL offers a fully automated close and consolidation solution for legal and management reporting, including support for Generally Accepted Accounting Principles (GAAP) and International Financial Reporting Standards (IFRS). Transactions are automatically processed and validated according to the best-practice business processes and control settings. In addition, an enterprise can proactively control expenditures by automatically checking spending requests against budget. With realtime reporting and information access, an enterprise can achieve complete visibility into financial results.

For more information about PeopleSoft GL, see the *PeopleSoft Enterprise General Ledger PeopleBook*.

1.4 Retail Sales Financial Business Process Flow

The Retail Sales Financial business process consists of the Post Channel Sales, Cash, and Deposits from Oracle ReSA to PeopleSoft GL integration flow.

[Figure 1–2](#) illustrates the Retail Sales Financial business process flow.

Figure 1–2 Retail Sales Financial Business Process Flow

Oracle ReSA sends summarized sales audit information to PeopleSoft GL for the Sales Journal. The sales audit information includes channel sales, cash, and deposits. The Oracle ReSA Export processes select and format corrected and previously audited data from the Oracle ReSA database so that it can be sent to PeopleSoft Financials.

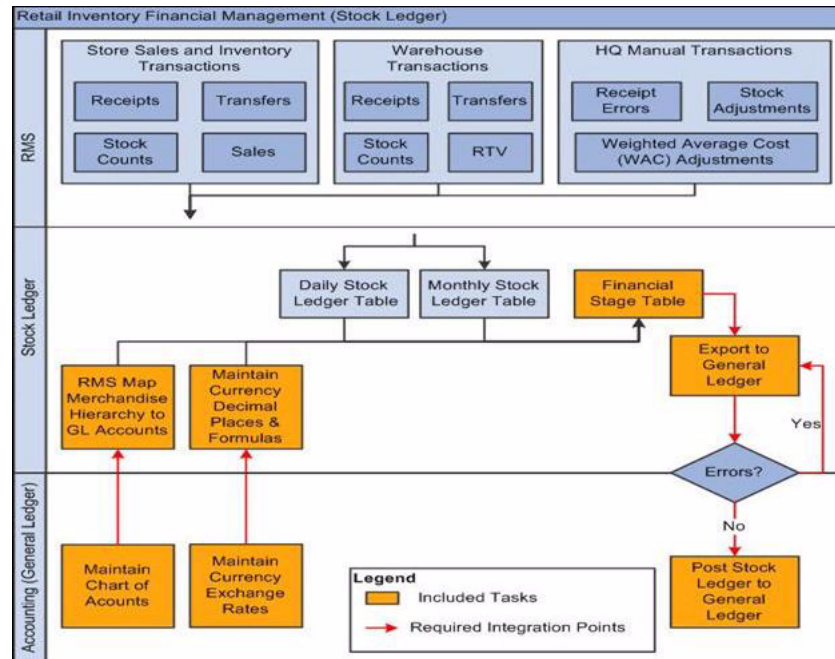
Oracle ReSA includes programs to automatically extract the required totals data and to format it to generic data files from a financial staging table for import into PeopleSoft GL. Sales audit data from Oracle ReSA is also posted directly to the Oracle RMS stock ledger and can be integrated into PeopleSoft GL through the stock ledger to the financial staging table and the accounting entry table. Before data is imported into PeopleSoft GL, a batch process writes balanced records to the financial staging table using the appropriate General Ledger account combinations (maintained in Cross Reference tables in Oracle ReSA and the Currency Exchange Rates).

The PeopleSoft Journal Generator loads the accounting entries from Oracle Retail to PeopleSoft GL journal entries. The accounting tables are referenced on the accounting entry definition defined for each type of accounting entry transaction. The Journal Generator uses the new accounting entry definitions to create PeopleSoft journal entries. After the journal entries are created by the PeopleSoft Journal Generator, they are edited and posted similarly to PeopleSoft subsystem journals.

1.5 Retail Inventory Financial Business Process Flow

The Retail Inventory Financial business process consists of the posting stock ledger from Oracle RMS to PeopleSoft GL. integration flow

Figure 1–3 illustrates the Retail Inventory Financial business process flow.

Figure 1-3 Oracle Retail Inventory Financials business process flow

Note: This diagram is for illustration only. Stores, warehouses, and corporate write other inventory related transactions to the Oracle RMS stock ledger.

The stock ledger in Oracle RMS records the financial results of the merchandising processes that occur in the Retail system, such as buying, selling, price changing, transferring, and so on. All of these transactions are recorded in the Oracle RMS stock ledger and rolled up to the subclass/location-level for days, weeks, and months. Daily and period-based financial information is scheduled to be loaded into PeopleSoft.

Oracle RMS sends three levels of stock ledger information to PeopleSoft GL:

- Monthly - no access to detailed reference information.
- Daily by subclass, class, or department - no access to detailed reference information.
- Daily by transaction.

The stock ledger transactions to be loaded into PeopleSoft are placed on the financial staging table with table triggers or batch, by the appropriate General Ledger account combinations (maintained in the Oracle RMS cross-reference table in Oracle Retail) and the currency exchange rates.

The PeopleSoft Journal Generator loads the accounting entries from Oracle Retail and creates PeopleSoft GL journal entries. The accounting tables are referenced on the accounting entry definition defined for each type of accounting entry transaction. The Journal Generator uses new accounting entry definitions to create PeopleSoft journal entries. After the journal entries are created by the Journal Generator, they are edited and posted similarly to PeopleSoft subsystem journals.

1.6 Retail Merchandise Procure to Pay Business Process Flow

The Retail Merchandise Procure to Pay business process consists of these integration flows:

- Post matched prepaid invoices from Oracle ReIM to PeopleSoft GL.
- Post manually matched paid invoices from Oracle ReIM to PeopleSoft GL.
- Post receipt write-offs from Oracle ReIM to PeopleSoft GL.
- Post matched invoices for payment from Oracle ReIM to PeopleSoft Payables.
- Post credit notes (matched or unmatched) for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post debit or credit memos for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post rebates for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post unmatched invoices for prepayment from Oracle ReIM to PeopleSoft Payables.

Figure 1-4, Figure 1-5, and Figure 1-6 illustrate the Retail Merchandise Procure to Pay business process flow:

Figure 1-4 Retail Merchandise Procure to Pay Business Process Flow (1 of 3)

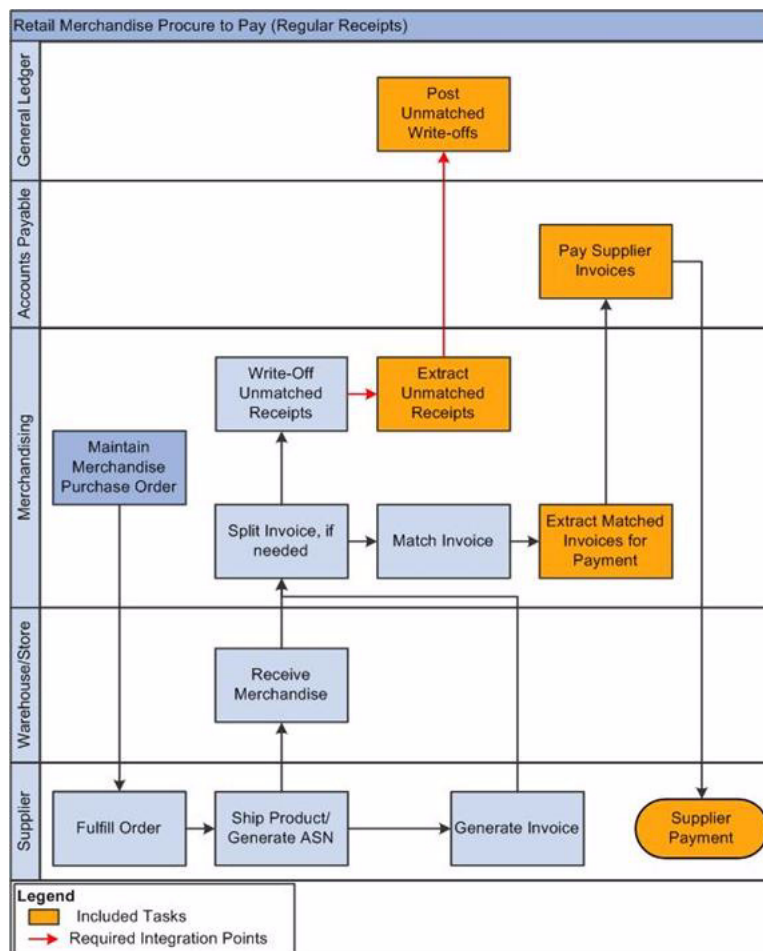


Figure 1–5 Retail Merchandise Procure to Pay business process flow (2 of 3)

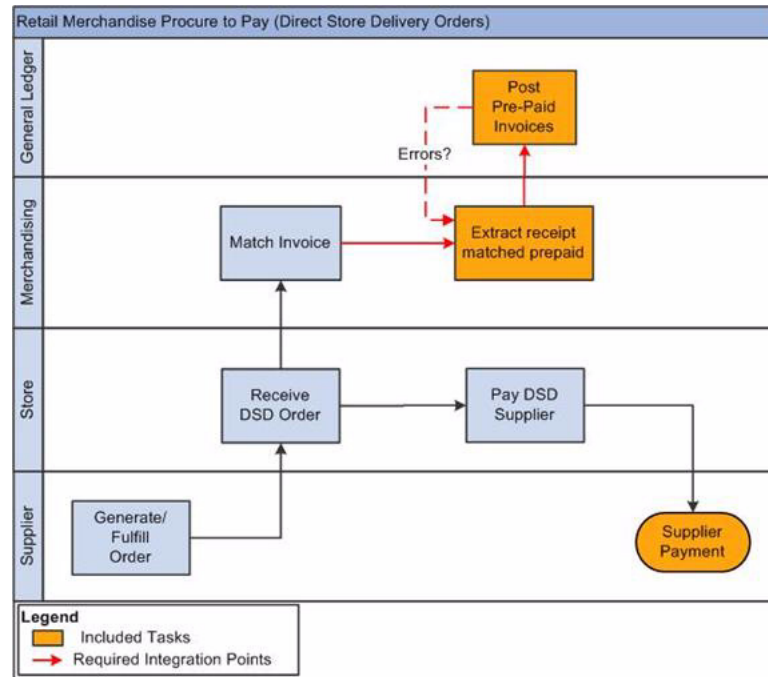
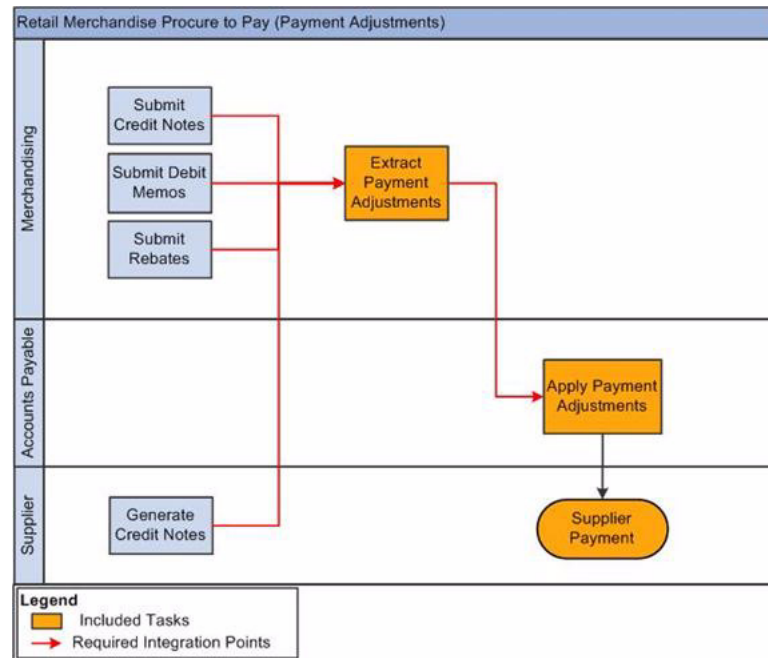


Figure 1–6 Retail Merchandise Procure to Pay business process flow (3 of 3)



The Retail Merchandise Procure to Pay business process flow enables posting of matched invoices, matched credit notes, debit and credit memos, rebates, and unmatched invoices for prepayment from Oracle ReIM to PeopleSoft Payables. It also facilitates drill back from PeopleSoft Payables to Oracle ReIM and drill forward from Oracle ReIM to PeopleSoft Payables.

1.7 Solution Assumptions and Constraints

These are the assumptions and constraints for this PIP:

PeopleSoft GL

1. You must implement the PeopleSoft applications before you implement this PIP.
2. The set of books in Oracle Retail is equivalent to the PeopleSoft GL business unit, ledger group, assuming the default ledger group and setID.

You are not required to map the ledger group and setID on the Oracle Retail side. PeopleSoft publishes the associated business units for each transaction that is setID-driven. You are not required to persist with the structure in the middle layer. Each transaction message accounts for the setID business unit association.

3. Oracle Retail manually creates and stores valid segment (ChartField) combinations in the appropriate GL cross-reference tables (Oracle ReSA, Oracle RMS, and Oracle ReIM).

4. For PeopleSoft, 16 ChartFields are available and 4 additional inactive ChartFields can be activated by customers. Oracle Retail supports 20 segments structure.

Therefore, the PeopleSoft Chart of Accounts can be made of up to 20 ChartFields (segments) only.

5. PeopleSoft ChartFields are 30 characters in length.

The Oracle Retail Integration Bus (RIB) subscription format supports only 25 characters for segments (ChartFields). Therefore, the segment length for Oracle Retail customers is 25 characters.

Currency Exchange Rate

1. The Oracle Retail stock ledger supports multiple currencies.

All transaction-level information is stored in the local currency of the store or warehouse where the transaction occurred. In Oracle ReIM, the invoice currency on the invoice is often the currency of the supplier, which can be different than the local currency.

2. The currency amounts precision is limited to 3 digits.

Calendar

1. The PeopleSoft applications use multiple calendars.

In this PIP, both PeopleSoft Financials and Oracle Retail use a single calendar.

2. After a calendar is established, this integration does not support switching the calendar.

Sales Audit/Stock Ledger Transactions

1. Oracle Retail sends the accounting date and the transaction date with its transactions.

These dates must not be changed or manipulated in PeopleSoft.

2. Users handle accounting entry errors manually on both the Oracle Retail and the PeopleSoft Financials side.

3. The system passes use or sales tax accounting information as part of the accounting entries between Oracle Retail and PeopleSoft Financials.

4. Oracle Retail calculates value-added tax (VAT).

PeopleSoft Journal Generator does not recalculate VAT. The system passes the VAT calculation as part of the accounting entry.

5. Oracle Retail stock ledger determines the valuation of inventory for merchandise being directly procured.

It passes this information to PeopleSoft as the accounting entries.

6. Oracle RMS, through the Oracle Retail stock ledger, provides PeopleSoft Financials with the value of ending inventory at cost using the method that the retailer indicates (cost method or retail method of accounting) by an adjusting entry.

This information is provided through monthly feed.

7. You cannot load accounting entries twice into PeopleSoft Financials.

When journals are created, a status field called *Distribution Status* changes to D; therefore, you cannot create new journal vouchers for the same set of accounting entries.

Other

1. Both PeopleSoft and Oracle Retail support multiple organizations in one application instance.
2. If you rename, change the name but not the value, or create new ChartFields (segments), you have customized the middle layer.
3. Oracle RIB is required for all message-based integration to Oracle Retail.

Constraints

1. Oracle Retail and PeopleSoft applications do not support modifying date by time zones.
2. PeopleSoft Financials can handle only three decimal places in amount fields.
For this integration, no currency in Oracle Retail must contain more than three decimal places. Otherwise, the amounts are rounded from four decimals to three decimals that may cause out-of-balance issues.
3. Customers switching from one financials application to another must re-implement the integration pack.
4. Oracle AIA uses the location ID as a key and, when it changes, Oracle AIA cannot associate it with the former ID from the create operation.

Therefore, PeopleSoft users experience an issue when they change the location ID of an existing supplier location in an update operation.

Note: Additional assumptions and constraints exist for each of the process integration flows; they are documented in their respective chapters.

Life Cycle Management for Reference Data

This chapter discusses the lifecycle management for reference data, including currency exchange rate, payment term, and suppliers information process integrations.

This chapter includes the following sections:

- [Section 2.1, "Currency Exchange Rate Integration"](#)
- [Section 2.2, "Payment Term Integration"](#)
- [Section 2.3, "Suppliers Information Integration"](#)

2.1 Currency Exchange Rate Integration

This section provides an overview of the process integration for initial loading and incremental synchronization of currency exchange rates between PeopleSoft General Ledger (GL) and the Oracle Retail Merchandising System (Oracle RMS) and discusses the following topics:

- [Section 2.1.2, "Currency Exchange Rate Integration Details"](#)
- [Section 2.1.3, "PeopleSoft GL Interfaces"](#)
- [Section 2.1.4, "Oracle RIB Interfaces"](#)
- [Section 2.1.5, "Core AIA Components"](#)
- [Section 2.1.6, "Integration Services"](#)

2.1.1 Currency Exchange Rate Integration Overview

Currency exchange rate is the reference information used in the translation of monetary values from one currency to another. The exchange rate expresses the value of one currency in terms of another. The process integration for currency exchange rates enables you to use PeopleSoft Financials as an accounting engine and Oracle Retail for sales audit and stock ledger transactions.

The process integration for currency exchange rates supports these integration flows:

- Load initial currency exchange rate from PeopleSoft GL to Oracle RMS
Enables the loading of all current effective-dated currency exchange rates from PeopleSoft GL to Oracle RMS for a new instance (logical or physical) of Oracle RMS
- Incremental creation and updates of currency exchange rates from PeopleSoft GL to Oracle RMS

Enables the synchronization of incremental creation and updates of the current effective-dated currency exchange rates from PeopleSoft GL to Oracle RMS

This integration is not a point-to-point integration between PeopleSoft GL and Oracle RMS. An Oracle Application Integration Architecture (Oracle AIA) layer serves as an intermediate thin layer of application between PeopleSoft GL and Oracle RMS. As part of the currency exchange rate integration, PeopleSoft GL sends the currency exchange rates to the Oracle AIA layer, and the Oracle AIA layer delivers the information to Oracle RMS. The Oracle AIA layer performs message filtering, message transformation, and message routing.

2.1.1.1 Solution Assumptions and Constraints

The integration design assumes that:

1. The PeopleSoft Financial Management system is responsible for adding and maintaining the currency exchange rates and types.
2. The PeopleSoft Financial Management system is the system of record.
3. The currency rate types are different in PeopleSoft and Oracle Retail applications.

The domain value maps (DVMs) are set up and maintained by both PeopleSoft GL and the Oracle AIA layer. Together, the COMMON value is found.

4. For initial synchronization, the Oracle Retail system date is set to match the PeopleSoft system date.

When initial synchronization is complete, Oracle Retail can restore the system date.

5. After the PeopleSoft currency exchange rates are published, the current effective-dated currency exchange rates are synchronized with the Oracle Retail exchange rates.
6. Oracle Retail accepts all records from PeopleSoft except the records with previous dates (before their vdate).
7. The PeopleSoft application maintains a table that stores effective-dated rate changes.
8. The PeopleSoft application supports triangulation of currency exchange rates, but Oracle Retail does not.

The PeopleSoft application converts its rates to a common rate before calling the web service for integration.

9. Retail RIB Error Hospital holds all the Oracle Retail side errors and handles any notification on its side.
10. If a value is not present for Oracle Retail in the CURRENCYEXCHANGE_CONVERSIONTYPECODE DVM, the Oracle AIA layer filters the record and does not send it to Oracle Retail.

The system sends a message to inform users that the value used for this DVM is not in the DVM. No error is generated.

11. If the value is missing for the CURRENCY_CODE DVM, the Oracle AIA layer passes the record to Oracle Retail with the PeopleSoft values.

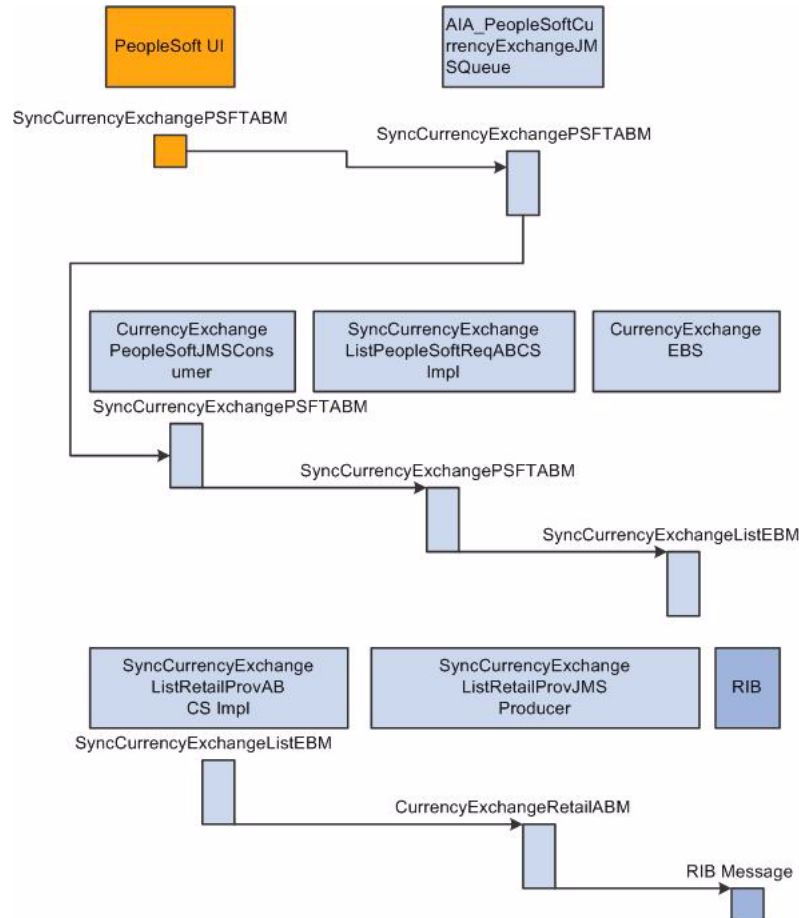
No error is generated.

12. Oracle Retail accepts only those currency exchange rates for which the from_currency is the base currency and ignores other exchange rates.

Oracle Retail handles this action internally. The AIA layer does not filter exchange rates.

Figure 2–1 illustrates the currency exchange rate integration flow.

Figure 2–1 Currency exchange rate integration flow



2.1.2 Currency Exchange Rate Integration Details

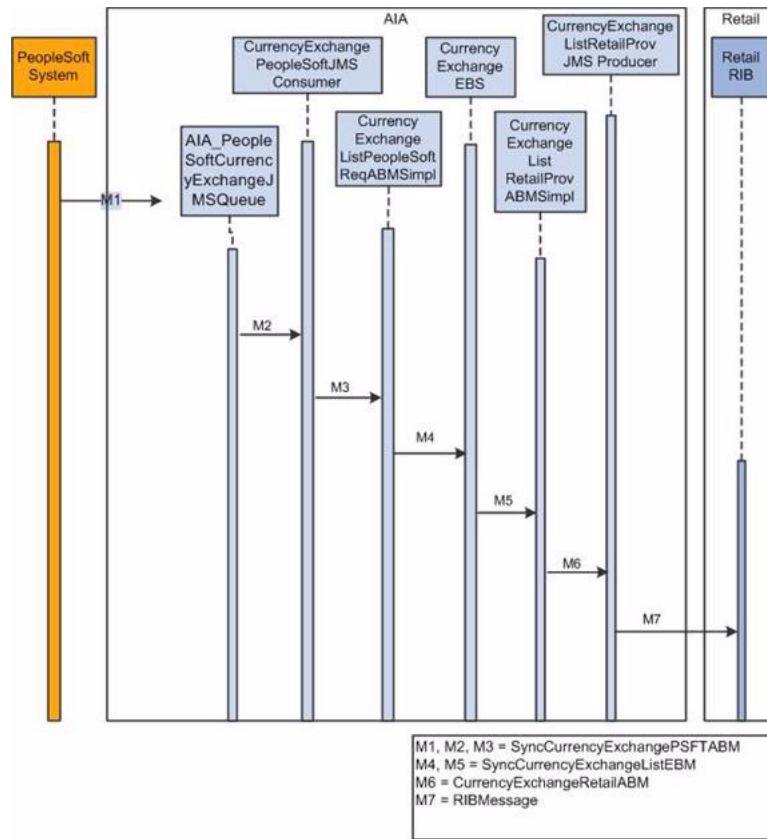
The integration flow uses these services:

- CurrencyExchangePeopleSoftJMSConsumer
- SyncCurrencyExchangeListPeopleSoftReqABCSImpl
- CurrencyExchangeEBS
- SyncCurrencyExchangeListRetailProvABCSImpl
- SyncCurrencyExchangeListRetailProvJMSProducer

2.1.2.1 Sequence Diagram

Figure 2–2 diagram illustrates the currency exchange rate integration flow.

Figure 2-2 Currency exchange rate integration sequence diagram



This sequence diagram is applicable to initial load, create, and update integration flows.

When you initiate the process:

1. PeopleSoft writes to the AIA_PeopleSoftCurrencyExchangeJMSQueue whenever a currency exchange rate is created or updated.
2. The CurrencyExchangePeopleSoftJMSConsumer service picks up the message from AIA_PeopleSoftCurrencyExchangeJMSQueue based on MessageName and invokes SyncCurrencyExchangeListPeopleSoftReqABCSImpl with SyncCurrencyExchangeListPSFTABM, and then this thin ReqABCS populates EBMHeader and invokes CurrencyExchangeEBS.
3. CurrencyExchangeEBS invokes SyncCurrencyExchangeListRetailProvABCSImpl or CAVS based on the routing rule.
4. The SyncCurrencyExchangeListRetailProvABCSImpl service transforms SyncCurrencyExchangeListEBM into Retail Application Business Message (ABM) and invokes SyncCurrencyExchangeListRetailProvJMSProducer.
5. The SyncCurrencyExchangeListRetailProvJMSProducer service transforms the CurrencyExchangeRetailABC to a RIBMessage and publishes the RIBMessage to the <username>.ETEXTCURRATE topic in the RIB Queue system.

Note: The structure of the PeopleSoft ABM is very close to that of an enterprise business message (EBM). PeopleSoft ABM does not have some details, such as the EBM header and the Oracle AIA namespace. These details are added by the PeopleSoft Application Business Connector Service (ABCS). Therefore, the PeopleSoft objects are referred as ABMs even though their content is structured in the EBM schema.

2.1.3 PeopleSoft GL Interfaces

Outbound Interactions

- SyncCurrencyExchangeList (Asynchronous)
- Request Schema: ExchangeRateSyncEBM.V1.xsd

For more information, see the *Enterprise PeopleTools PeopleBook*, "PeopleSoft Integration Broker," Managing Service Operations.

2.1.4 Oracle RIB Interfaces

Inbound Interactions

- SyncCurrencyExchangeList(Asynchronous)
- Request Schema: CurrRateDesc.xsd

For more information, see the *RIB Integration Guide*.

2.1.5 Core AIA Components

The currency exchange rate integration uses these components:

- CurrencyExchangeEBO
- CurrencyExchangeEBM
- CurrencyExchangeEBS

The core enterprise business object (EBO) and EBM XSD files can be located by EBO within this parent folder: \$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core Enterprise Business Service (EBS) WSDL files can be located by EBO within this parent folder: \$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository.

For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

You can extend EBOs, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Extensibility for Oracle AIA Artifacts," Extending EBOs.

2.1.6 Integration Services

These services are delivered with the process integration for currency exchange rate:

- CurrencyExchangePeopleSoftJMSConsumer
- SyncCurrencyExchangeListPeopleSoftReqABCImpl
- CurrencyExchangeEBS
- SyncCurrencyExchangeListRetailProvABCImpl
- SyncCurrencyExchangeListRetailProvJMSProducer

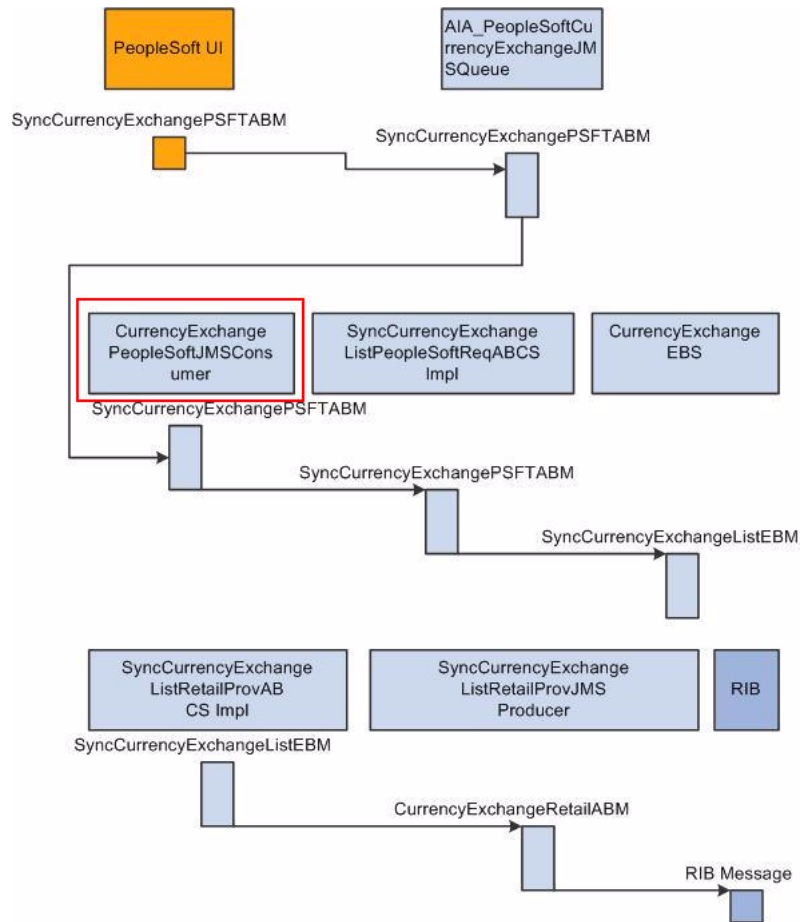
For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

2.1.6.1 CurrencyExchangePeopleSoftJMSConsumer

CurrencyExchangePeopleSoftJMSConsumer is a Mediator service. It has a JMS adapter called SyncCurrencyExchangeListPeopleSoftJMSConsumer. This adapter listens to the AIA_PeopleSoftCurrencyExchangeJMSQueue and picks up the messages for which MessageName is SyncCurrencyExchangeList. The system invokes SyncCurrencyExchangeListPeopleSoftReqABCImpl with the SyncCurrencyExchangeListPSFTABM.

[Figure 2–3](#) illustrates the relationship of the CurrencyExchangePeopleSoftJMSConsumer with the other services in the integration flow.

Figure 2-3 CurrencyExchangePeopleSoftJMSConsumer relationship in the integration flow

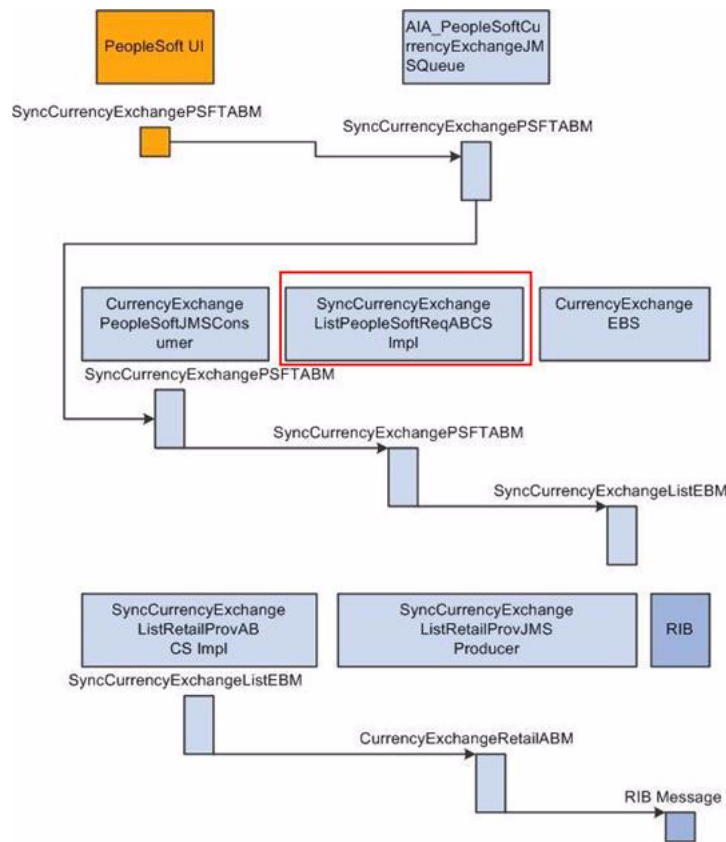


2.1.6.2 SyncCurrencyExchangeListPeopleSoftReqABCImpl

The SyncCurrencyExchangeListPeopleSoftReqABCImpl service is a BPEL process. This service is a thin requester that receives SyncCurrencyExchangeListPSFTABM from SyncCurrencyExchangePeopleSoftConsumer. This message is transformed to SyncCurrencyExchangeListEBM, which populates EBM Header and populates the cross-reference table. No DVM lookups are required for this message because the PeopleSoft system sends all the Oracle AIA Common values. The system sends this SyncCurrencyExchangeListEBM to CurrencyExchangeEBS.

Figure 2-4 illustrates the relationship of the SyncCurrencyExchangeListPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–4 SyncCurrencyExchangeListPeopleSoftReqABCImpl relationship in the integration flow

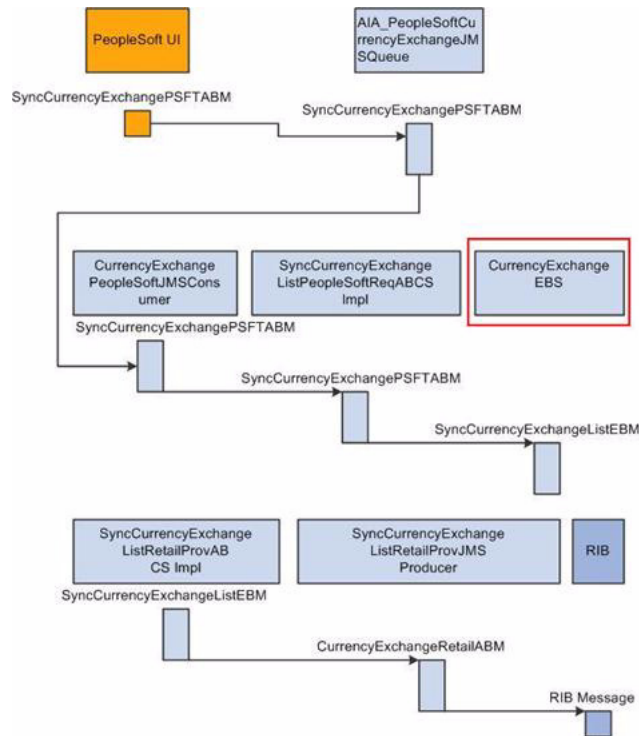


2.1.6.3 CurrencyExchangeEBS

CurrencyExchangeEBS routes all currency exchange related actions, such as create and update of currency exchange rate as Sync Currency Exchange Rate message to SyncCurrencyExchangeListRetailProvABCImpl or Composite Application Validation System (CAVS) based on the filter condition and operations. For this particular integration, only the Sync action is available. Updates and creates are done using the Sync action. Oracle Retail determines whether this Sync Currency Exchange Rate message is for a create action or an update action.

Figure 2–5 illustrates the relationship of the CurrencyExchangeEBS with the other services in the integration flow.

Figure 2–5 CurrencyExchangeEBS relationship in the integration flow



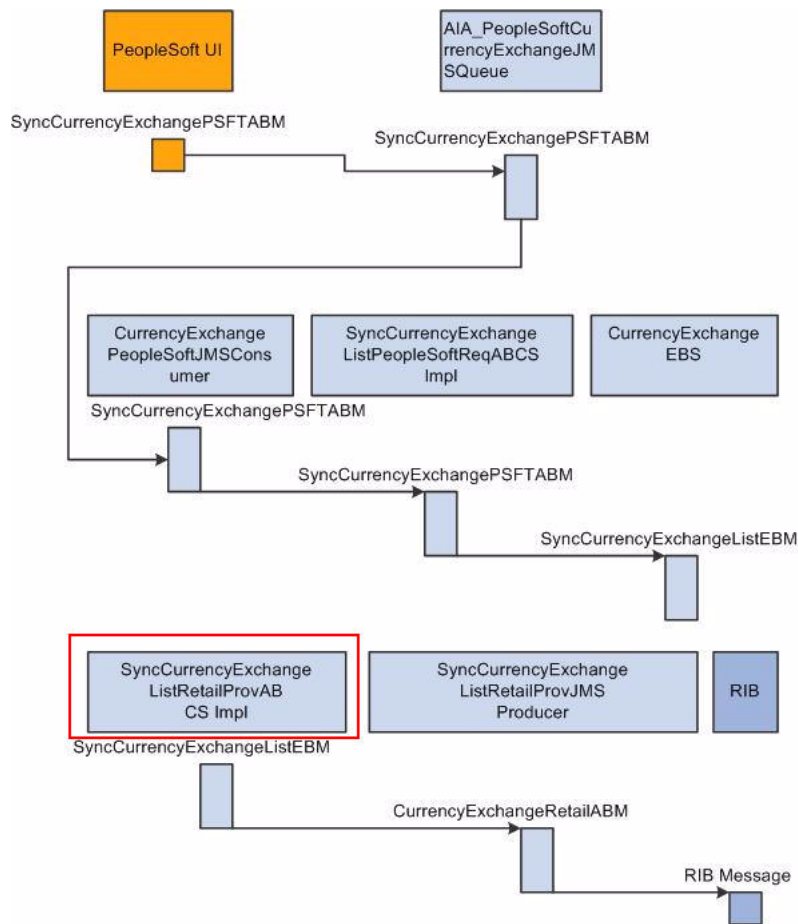
For more information about this EBS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Developing EBSs" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding EBSs."

2.1.6.4 SyncCurrencyExchangeListRetailProvABCSImpl

SyncCurrencyExchangeListRetailProvABCSImpl is a BPEL process. This service receives the SyncCurrencyExchangeListEBM as a request from CurrencyExchangeEBS and transforms it to the Retail ABM.

Figure 2–6 illustrates the relationship of the SyncCurrencyExchangeListRetailProvABCSImpl with the other services in the integration flow.

Figure 2–6 SyncCurrencyExchangeListRetailProvABCImpl relationship in the integration flow



2.1.6.5 SyncCurrencyExchangeListRetailProvJMSProducer

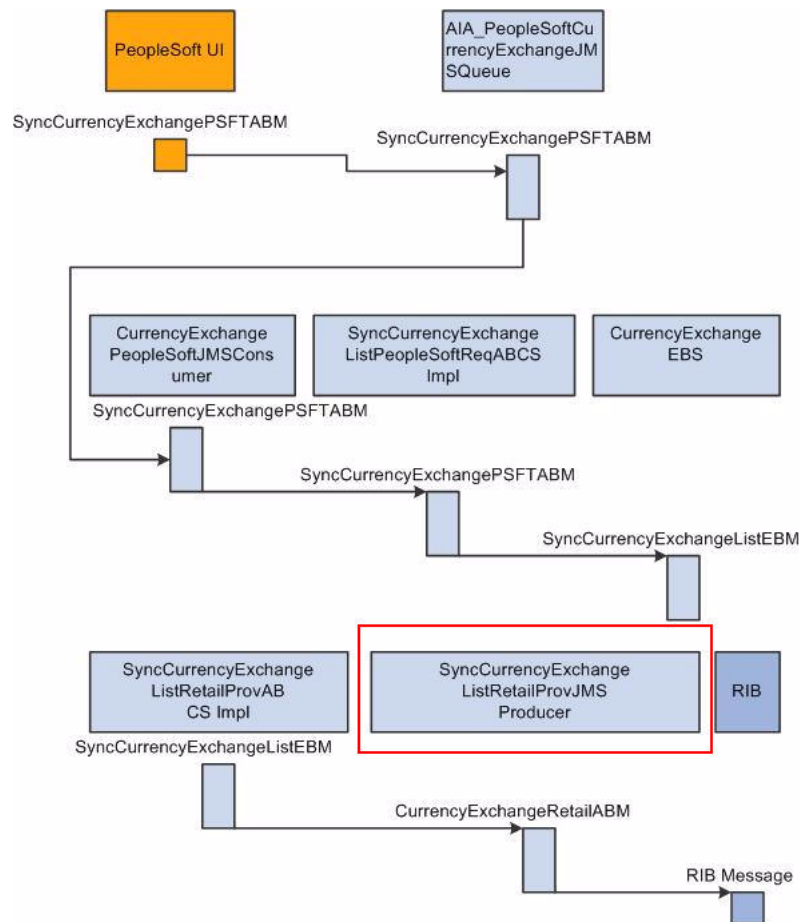
SyncCurrencyExchangeListRetailProvJMSProducer is a BPEL process. This service is responsible for taking the SyncCurrencyExchangeListRetailABM and transforming it into the RIBMessage format and then putting the message into the <username>.ETEXTCURRATE topic in the RIB Queue system.

This service performs these actions:

- Accepts the CurrencyExchangeRetailABM from SyncCurrencyExchangeListRetailProvABCImpl
- Transforms CurrencyExchangeRetailABM into a RIB message
- Puts the RIB message into the <username>.ETEXTCURRATE topic in the RIB Queue system.

Figure 2–7 illustrates the relationship of the SyncCurrencyExchangeListRetailProvJMSProducer with the other services in the integration flow.

Figure 2–7 SyncCurrencyExchangeListRetailProvJMSProducer relationship in the integration flow



2.2 Payment Term Integration

This section provides an overview of the process integration for initial loading and incremental synchronization of payment terms between PeopleSoft Payables and the Oracle RMS and discusses the following topics:

- [Section 2.2.2, "Payment Term Integration Details"](#)
- [Section 2.2.3, "PeopleSoft Payables Interfaces"](#)
- [Section 2.2.4, "Oracle RMS Interfaces"](#)
- [Section 2.2.5, "Core AIA Components"](#)
- [Section 2.2.6, "Integration Services"](#)

2.2.1 Payment Term Integration Overview

In the integrated environment, PeopleSoft Enterprise Financials acts as a payable and accounting engine with Oracle RMS for supplier payment, merchandise write-offs, and prepaid adjustments. It eliminates the need for manually reentering reference data from PeopleSoft Payables to Oracle RMS. Benefits to retailers include reducing the labor cost of double entry and providing more accurate and effective payment of invoices, payment adjustments, and accounting records.

PeopleSoft Payables is the source of valid payment terms. Oracle RMS uses payment terms to apply the correct payment terms to a supplier or purchase order and ensures correct timing of payment and application of payment term discounts.

The payment terms integration synchronizes payment terms information from PeopleSoft Payables to Oracle RMS through these integration flows:

- Load initial payment term from PeopleSoft Payables to Oracle RMS. Enables the loading of all current effective-dated payment terms from PeopleSoft Payables to Oracle RMS for a new instance (logical or physical) of Oracle RMS.
- Incremental creation and updates of current effective dated payment term from PeopleSoft Payables to Oracle RMS. Enables the synchronization of incremental creation and updates of the payment terms from PeopleSoft Payables to Oracle RMS.

For more information about payment terms, see *PeopleSoft Enterprise Source to Settle Common Information PeopleBook*, "Defining Procurement Options," Defining Payment Terms.

This integration is not a point-to-point integration between PeopleSoft Payables and Oracle RMS. An Oracle AIA layer serves as an intermediate thin layer of application between PeopleSoft Payables and Oracle RMS. As part of the payment term integration, PeopleSoft Payables sends the payment term to the Oracle AIA layer, and the Oracle AIA layer delivers the information to Oracle RMS. The Oracle AIA layer performs message filtering, message transformation, and message routing.

2.2.1.1 Solution Assumptions and Constraints

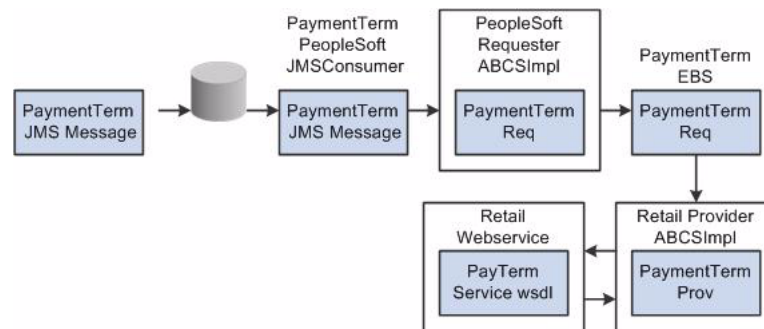
The integration design assumes that:

1. Oracle Retail can handle only single-tier payment terms.
PeopleSoft Payables supports multiple-tier payment terms for installment payments.
2. If the PeopleSoft Payables inactivates a payment term and the end date is not before the current system date or vdate in Oracle Retail, then Oracle Retail rejects it.
3. All Oracle Retail business units have the same set of payment terms.
4. Only the *Sync* operation accepts a list (or collection) message.
The *Create* and *Update* operations only accept a single message for new rows created or updated in PeopleSoft Payables.
5. PeopleSoft Payables can have the same payment terms code in different setIDs.
However, Oracle Retail only supports global payment terms. PeopleSoft sends a common value that represents the setID/payment terms code combination. Users should not set up the same payment terms code value under multiple setIDs when using this integration.
6. For *Sync*, *Create*, and *Update* payment term messages, PeopleSoft sends GUID as the PeopleSoft key for the payment term header and the terms sequence number for payment term lines.
7. PeopleSoft payment terms have a record in the Oracle Retail language before you run the *Sync* operation.
The last update to a payment term in the PeopleSoft application should be in the Oracle Retail language so that the translatable fields appear correct in Oracle Retail.

8. Oracle Retail does not allow users to create and update payment terms in Oracle RMS.
9. Payment term integration occurs before supplier initial load and manual setup of freight term in Oracle RMS.
10. The `CreatePaymentTermRetailProvABCImpl`, `SyncPaymentTermListRetailProvABCImpl`, and `UpdatePaymentTermRetailProvABCImpl` assign the Creation Date Time, originating from the PeopleSoft ABCS implementation, to the `end_date_Active` if the status code from PeopleSoft is `I`.

Figure 2–8 illustrates the payment term integration flow.

Figure 2–8 Payment term integration flow



2.2.2 Payment Term Integration Details

These services are common to *Create*, *Update*, and *Sync* payment term integration flows:

- `PaymentTermPeopleSoftJMSConsumer`
- `PaymentTermEBS`
- `PayTermService`

These services are specific to *Sync* payment term integration flow:

- `SyncPaymentTermListPeopleSoftReqABCImpl`
- `SyncPaymentTermListRetailProvABCImpl`

These services are specific to *Create* payment term integration flow:

- `CreatePaymentTermPeopleSoftReqABCImpl`
- `CreatePaymentTermRetailProvABCImpl`

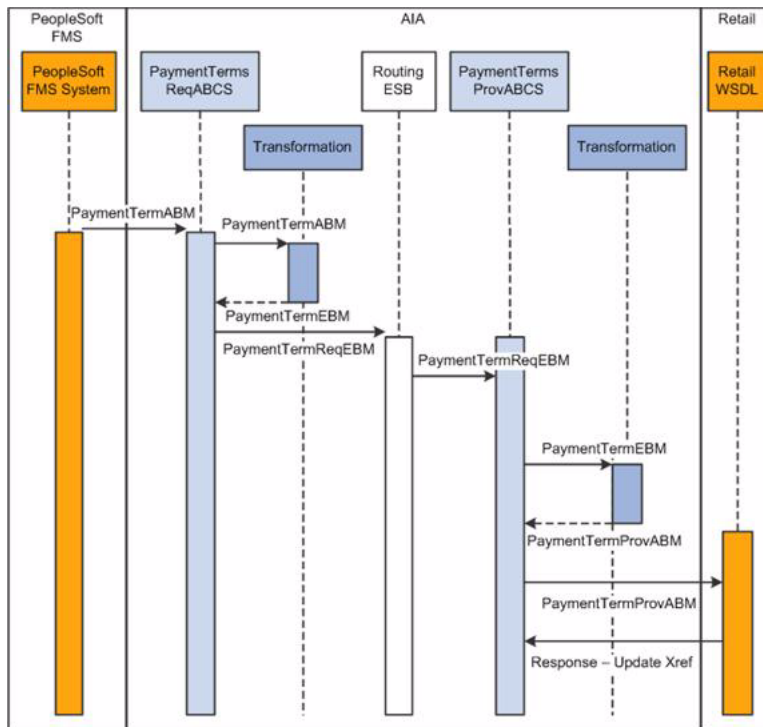
These services are specific to *Update* payment term integration flow:

- `UpdatePaymentTermPeopleSoftReqABCImpl`
- `UpdatePaymentTermRetailProvABCImpl`

2.2.2.1 Sequence Diagram

Figure 2–9 illustrates the synchronize payment term sequence flow.

Figure 2–9 Synchronize payment term sequence flow diagram



This sequence diagram is applicable for initial load, create, and update integration flows.

When you initiate the process:

1. PeopleSoft puts the SyncPaymentTermListPSFTABM, CreatePaymentTermPSFTABM, and UpdatePaymentTermPSFTABM in the AIA_PeopleSoftPaymentTermsJMSQueue with MessageName set to SyncPaymentTermList/CreatePaymentTerm/UpdatePaymentTerm.
2. The PaymentTermPeopleSoftJMSConsumer listens on the AIA_PeopleSoftPaymentTermsJMSQueue and invokes SyncPaymentTermListPeopleSoftReqABCSImpl if SyncPaymentTermListPSFTABM, CreatePaymentTermPeopleSoftReqABCSImpl if CreatePaymentTermPSFTABM, and UpdatePaymentTermPeopleSoftReqABCSImpl if UpdatePaymentTermPSFTABM.
3. For the initial load, PeopleSoft invokes the SyncPaymentTermListPeopleSoftReqABCSImpl service with SyncPaymentTermListPSFTABM.

This service performs these actions:

- a. Transforms SyncPaymentTermListPSFTABM to SyncPaymentTermListEBM.
- b. Populates EBMHeader.
- c. Updates cross-reference data.
- d. Enriches the message and puts action code for header and lines if it is *Create* or *Update*, and invokes PaymentTermEBS.
4. When you create a new payment term in PeopleSoft, the system invokes CreatePaymentTermPeopleSoftReqABCSImpl with CreatePaymentTermPSFTABM.

This service performs these actions:

- a. Transforms CreatePaymentTermPSFTABM to CreatePaymentTermEBM.
 - b. Populates EBMHeader.
 - c. Updates cross-reference data.
 - d. Invokes PaymentTermEBS.
5. When you update a payment term in PeopleSoft, the system invokes the UpdatePaymentTermPeopleSoftReqABCServiceImpl service with UpdatePaymentTermPSFTABM.

This service performs these actions:

- a. Transforms UpdatePaymentTermPSFTABM to UpdatePaymentTermEBM.
 - b. Populates EBMHeader.
 - c. Enriches the message and puts action code for lines if it is *Create* or *Update* and invokes PaymentTermEBS.
6. PaymentTermEBS invokes SyncPaymentTermListRetailProvABCServiceImpl, CreatePaymentTermRetailProvABCServiceImpl, UpdatePaymentTermRetailProvABCServiceImpl, or CAVS based upon the operation and filter condition.
7. The SyncPaymentTermRetailProvABCServiceImpl service transforms SyncPaymentTermListEBM to Retail ABM.

Transformation synchronously invokes create or update PayTermService web service from Oracle RMS.

8. The CreatePaymentTermRetailProvABCServiceImpl service transforms CreatePaymentTermEBM to Retail ABM.

Transformation synchronously invokes the create PayTermService web service from Oracle RMS.

9. The UpdatePaymentTermRetailProvABCServiceImpl transforms UpdatePaymentTermEBM to Retail ABM.

Transformation synchronously invokes the update PayTermService web service from Oracle RMS.

Note: The structure of the PeopleSoft ABM is very close to that of an EBM. PeopleSoft ABM does not have some details, such as the EBM header and the Oracle AIA namespace. These details are added by the PeopleSoft ABCS. Therefore, the PeopleSoft objects are referred as ABMs even though they have the content structured in the EBM schema.

2.2.2.2 Multiple Language Support

Oracle Retail supports multiple languages, however, one language is considered primary.

The PeopleSoft system supports several languages per instance based on user localization preferences. Any translatable data attribute (such as payment term name) may have multiple occurrences, each one representing the language of the user who entered or updated it in the PeopleSoft system. For example, the data field for the

payment term name has a French entry from a user with a French UI preference and an English entry from a user with an English UI preference.

Because Oracle Retail supports only one primary language per instance, the Oracle AIA layer filters Oracle Retail records on the provider based on the EBM header language code. The primary language supported by the Oracle Retail instance is identified in `AIAConfigurationProperties.XML`.

The filtering operation includes a DVM language code lookup based on the common EBM Header. If the DVM language code lookup returns the same value stored in the `AIAConfigurationProperties.XML`, then the Oracle Retail record on the provider is accepted.

Constraints

While PeopleSoft Payables can support multiple-tier payment terms, Oracle RMS can support only single-tier payment terms. The Oracle AIA integration for payment terms passes only translatable fields from PeopleSoft to Oracle Retail for the primary language of the Oracle Retail instance being integrated. This condition applies to the incremental creates and updates of payment terms.

For the incremental creates and updates of payment terms, Oracle AIA does not filter the message from PeopleSoft. Oracle AIA sends the message in the same language in which it was created or updated, even if this language is not the primary language of the Oracle Retail instance.

Oracle AIA on the Oracle Retail side filters initial load, create, and update messages and sends only those messages that have the language code similar to the Oracle Retail primary language code.

2.2.3 PeopleSoft Payables Interfaces

Outbound Interactions

- SyncPaymentTerm (Asynchronous, one-way)
Request Schema: AP_PAY_TERM_SYNC_EBM.V1.xsd
- CreatePaymentTerm (Asynchronous, one-way)
Request Schema: AP_CREATE_PAY_TERM_SYNC_EBM.V1.xsd
- UpdatePaymentTerm (Asynchronous, one-way)
Request Schema: AP_UPDATE_PAY_TERM_SYNC_EBM.V1.xsd

2.2.4 Oracle RMS Interfaces

Inbound Interactions

- PayTermService
 - Operation: CreatePayTermDesc, Request Schema: PayTermDesc.xsd, Response Schema: PayTermRef.xsd
 - Operation: UpdatePayTermDesc, Request Schema: PayTermDesc.xsd, Response Schema: PayTermRef.xsd
 - Operation: CreateDetailPayTermDesc, Request Schema: PayTermDesc.xsd, Response Schema: PayTermRef.xsd
 - Operation: UpdateHeaderPayTermDesc, Request Schema: PayTermDesc.xsd, Response Schema: PayTermRef.xsd

For more information, see the *RMS Operations Guide*, "Web Services Overview."

2.2.5 Core AIA Components

The Payment term integration uses these components:

- PaymentTermEBO
- PaymentTermEBS
- PaymentTermEBM

The core EBO and EBM XSD files can be located by EBO within this parent folder:

\$AIA_

HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core EBS WSDL files can be located by EBO within this parent folder: \$AIA_

HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation about individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in the Oracle Enterprise Repository.

For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

EBOs can be extended, for instance, to add new data elements. These extensions are protected, and remain intact after a patch or an upgrade.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Extensibility for AIA Artifacts," Extending EBOs.

2.2.6 Integration Services

These services are delivered with the process integration for payment term:

- PaymentTermPeopleSoftJMSConsumer
- SyncPaymentTermListPeopleSoftReqABCImpl
- CreatePaymentTermPeopleSoftReqABCImpl
- UpdatePaymentTermPeopleSoftReqABCImpl
- PaymentTermEBS
- SyncPaymentTermListRetailProvABCImpl
- CreatePaymentTermRetailProvABCImpl
- UpdatePaymentTermRetailProvABCImpl
- PayTermService

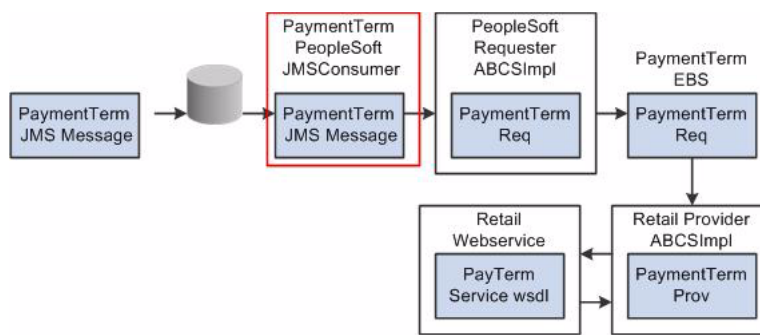
For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

2.2.6.1 PaymentTermPeopleSoftJMSConsumer

This service is a Mediator process. It has three different JMS adapters: SyncPaymentTermListPeopleSoftJMSConsumer, CreatePaymentTermPeopleSoftJMSConsumer, and UpdatePaymentTermPeopleSoftJMSConsumer. Each of these adapters listens to the AIA_PeopleSoftPaymentTermsJMSQueue. When a message is found in the JMS queue, based on MessageName, one of three adapters picks up the message and invokes SyncPaymentTermListPeopleSoftReqABCSImpl, CreatePaymentTermPeopleSoftReqABCSImpl, or UpdatePaymentTermPeopleSoftReqABCSImpl, respectively.

Figure 2–10 illustrates the relationship of the PaymentTermPeopleSoftJMSConsumer with the other services in the integration flow.

Figure 2–10 Relationship of PaymentTermPeopleSoftJMSConsumer to other services



2.2.6.2 SyncPaymentTermListPeopleSoftReqABCSImpl

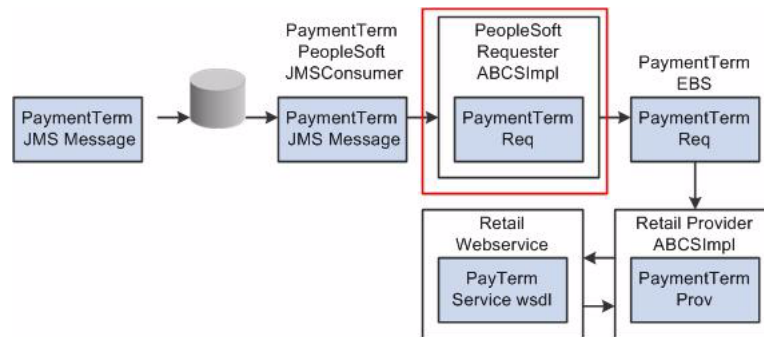
SyncPaymentTermListPeopleSoftReqABCSImpl is a thin requester that receives SyncPaymentTermListPSFTABM from PeopleSoft primarily for synchronizing the initial loads. This is a single operation service that has PaymentTermEBS as a partner service. It transforms the input to SyncPaymentTermListEBM, which is primarily for populating the EBM Header, populating the cross-reference, and setting the environment code to CAVS or PRODUCTION based on the RouteToCAVS property in the AIAConfigurationProperties file. The Request message adheres to the SyncPaymentTermListEBM XML Schema element.

This service is a BPEL process.

Partner Link Service: The SyncPaymentTermPeopleSoftReqABCSImplExt is an extensibility service that enables customers to filter, validate, or augment the input. This service accepts SyncPaymentTermListEBM as input and returns the same.

Figure 2–11 illustrates the relationship of the SyncPaymentTermListPeopleSoftReqABCSImpl with the other services in the integration flow.

Figure 2–11 Relationship of SyncPaymentTermListPeopleSoftReqABCImpl to other services



2.2.6.3 CreatePaymentTermPeopleSoftReqABCImpl

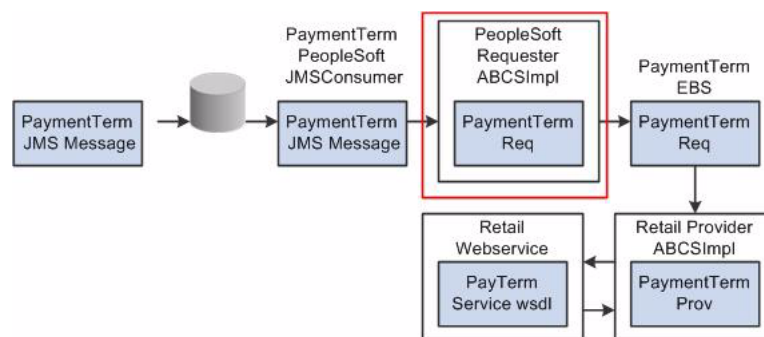
CreatePaymentTermPeopleSoftReqABCImpl is a thin requester that receives CreatePaymentTermPSFTABM from PeopleSoft. This input is transformed to CreatePaymentTermEBM. It populates EBM Header cross-reference data population and sends this EBM to PaymentTermEBS.

This is a single operation service that has PaymentTermEBS as a partner service. This service is a BPEL process.

Partner Link Service: The CreatePaymentTermPeopleSoftReqABCImplExtension is an extensibility service that enables customers to filter, validate, or augment the input. This service accepts CreatePaymentTermEBM as input and returns the same.

Figure 2–12 illustrates the relationship of the CreatePaymentTermPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–12 Relationship of CreatePaymentTermPeopleSoftReqABCImpl to other services



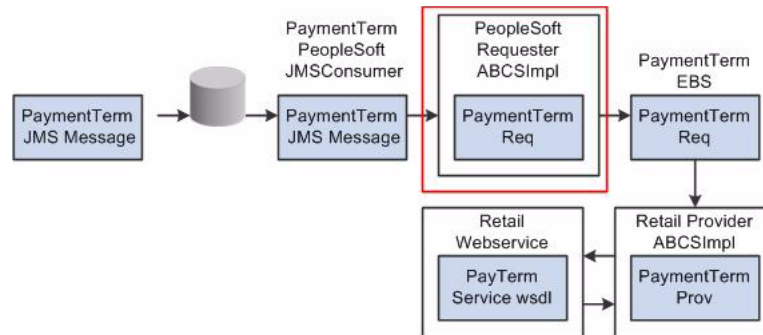
2.2.6.4 UpdatePaymentTermPeopleSoftReqABCImpl

UpdatePaymentTermPeopleSoftReqABCImpl is a thin requester that receives UpdatePaymentTermPSFTABM from PeopleSoft. It transforms the input to UpdatePaymentTermEBM. It also populates EBM Header and sends this EBM to PaymentTermEBS. This is a single operation service, which has PaymentTermEBS as a partner service.

Partner Link Service: The UpdatePaymentTermPeopleSoftReqABCImplExt is an extensibility service that enables customers to filter, validate, or augment the input. This service accepts UpdatePaymentTermEBM as input and returns the same.

Figure 2–13 illustrates the relationship of the UpdatePaymentTermPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–13 Relationship of UpdatePaymentTermPeopleSoftReqABCImpl to other services

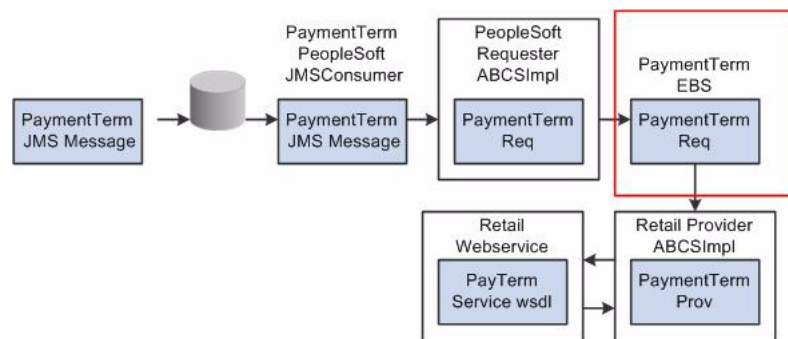


2.2.6.5 PaymentTermEBS

PaymentTermEBS is a Mediator service. It routes to SyncPaymentTermListRetailProvABCImpl, CreatePaymentTermRetailProvABCImpl, UpdatePaymentTermRetailProvABCImpl, or CAVS based on the operation and filter condition.

Figure 2–14 illustrates the relationship of the PaymentTermEBS with the other services in the integration flow.

Figure 2–14 Relationship of PaymentTermEBS to other services



For more information about this EBS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Developing EBSs" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding EBSs."

2.2.6.6 SyncPaymentTermListRetailProvABCImpl

SyncPaymentTermListRetailProvABCImpl is a BPEL process. This service accepts a SyncPaymentTermListEBM message as input from PaymentTermEBS. It transforms the input to PaymentTermABM and invokes the web service PayTermService.

This SyncPaymentTermEBM undergoes one of four transformations that transform the input to four types of messages that Oracle Retail accepts, based on the header and detail (line) action code. While transforming the data, the DVM looks up the common EBM language and common status codes to determine the RMS values.

Table 2–1 lists the operations used by this service and related messages.

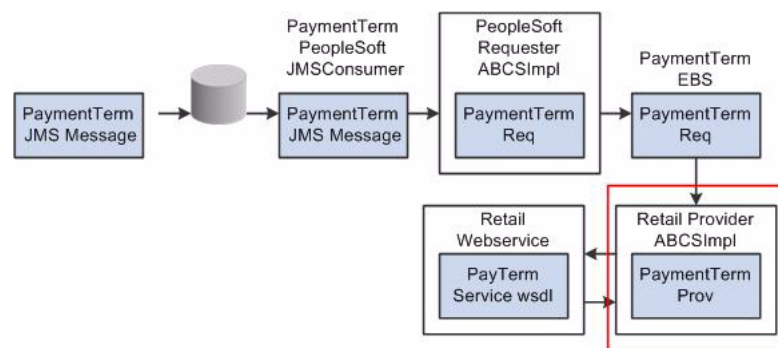
Table 2–1 Operations Used by SyncPaymentTermListRetailProvABCImpl

Operation	Input Message	Output Message
CreatePayTermDesc	PayTermDesc	PayTermRef
UpdateHeaderPayTermDesc	PayTermDesc	PayTermRef
CreateDetailPayTermDesc	PayTermDesc	PayTermRef
UpdatePayTermDesc	PayTermDesc	PayTermRef

The response message from Retail PayTermService for the operation corresponding to PayTermDtlCre message has the *terms-seq*, which you should use to populate the RETL column in the cross-reference table PAYMENTTERMLINE_ID.

Table 2–15 illustrates the relationship of the SyncPaymentTermListRetailProvABCImpl with the other services in the integration flow.

Figure 2–15 Relationship of SyncPaymentTermListRetailProvABCImpl to other services



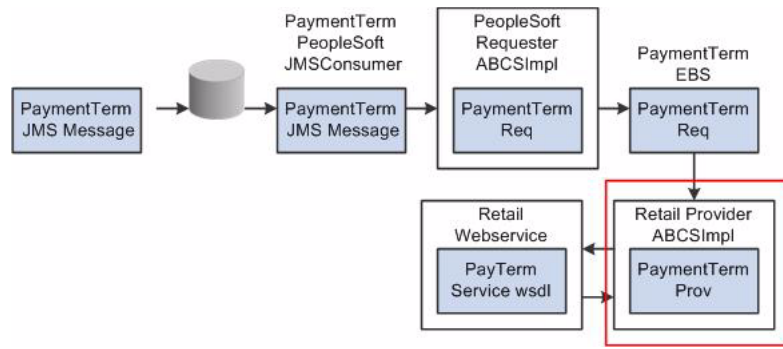
2.2.6.7 CreatePaymentTermRetailProvABCImpl

CreatePaymentTermRetailProvABCImpl is a BPEL process. It accepts CreatePaymentTermEBM as input from PaymentTermEBS. It then sends this PaymentTermABM to the PreProcessABM operation of CreatePaymentTermRetailProvABCImplExt if a service property is set to *true* in the AIAConfigurationProperties file. This extension service returns the same CreatePaymentTermEBM after processing it, and then transforms it into a PayTermCre message type PaymentTermABM. The PayTermService operation CreatePayTermDesc is invoked with this PaymentTermABM.

The response message from Retail PayTermService, for the operation corresponding to PayTermDtlCre message, has the *terms-seq*, which you should use to populate the RETL column in the cross-reference table PAYMENTTERMLINE_ID. While transforming the data, the DVM looks up the common status code to determine the RMS-enabled flag value.

Table 2–16 illustrates the relationship of the CreatePaymentTermRetailProvABCImpl with the other services in the integration flow.

Figure 2–16 Relationship of CreatePaymentTermRetailProvABCImpl to other services



2.2.6.8 UpdatePaymentTermRetailProvABCImpl

UpdatePaymentTermRetailProvABCImpl is a BPEL process. It accepts UpdatePaymentTermEBM as input from PaymentTermEBS. This EBM invokes the PreProcessEBM operation of UpdatePaymentTermRetailProvABCImplExt if a service property in the AIAConfigurationProperties file is set to *true*. This extension service returns the same UpdatePaymentTermEBM after processing it.

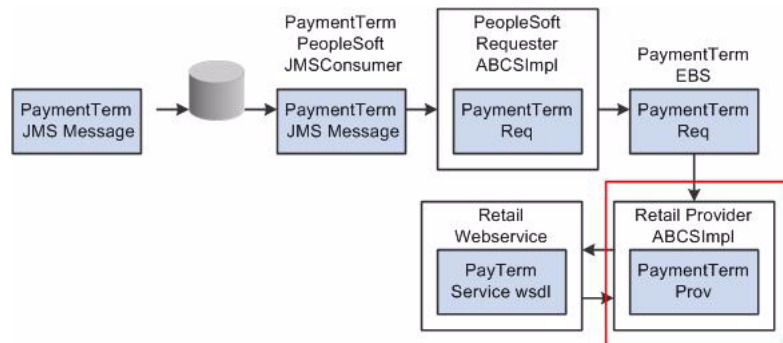
This UpdatePaymentTermEBM undergoes one of three transformations that transform the input to PayTermMod (updateHeaderPayTermDesc), PayTermDtlMod (updatePayTermDesc), or PayTermDtlCre (createDetailPayTermDesc) message type PaymentTermEBS.

After dynamically assigning the partner link based on the RouteToCAVS property in the AIAConfig file, it invokes the Retail PayTermService web service or CAVS. The corresponding operation of the PayTermService is invoked with the corresponding message type Retail ABMs if the RouteToCAVS property is set to *false*.

The response message from Retail PayTermService for an operation corresponding to the PayTermDtlCre message has the *terms-seq*, which you should use to populate the RETL column in the cross-reference table PAYMENTTERMLINE_ID.

Table 2–17 illustrates the relationship of the UpdatePaymentTermRetailProvABCImpl with the other services in the integration flow.

Figure 2–17 Relationship of UpdatePaymentTermRetailProvABCImpl to other services



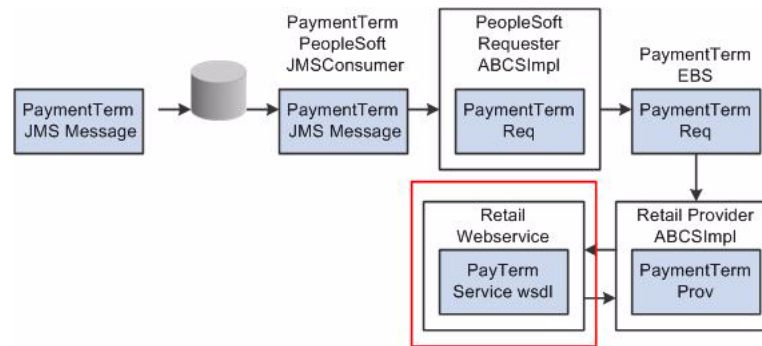
2.2.6.9 PayTermService

PayTermService accepts PaymentTermABM message types for operations createPayTermDesc, updateHeaderPayTermDesc, updatePayTermDesc, and createDetailPayTermDesc from SyncPaymentTermListRetailProvABCImpl (for synchronization of a bulk of updated payment terms data) or

UpdatePaymentTermRetailProvABCSImpl (for an update of an old payment term). This web service is called synchronously and returns the Retail ID for cross-reference. The PayTermService cannot accept collections, so the messages are split before being sent. While transforming the data, the DVM looks up the common status code to determine the RMS-enabled flag value.

- [Table 2–18](#) illustrates the relationship of the PayTermService with the other services in the integration flow.

Figure 2–18 Relationship of PayTermService to other services



2.3 Suppliers Information Integration

This section provides an overview of the process integration for initial loading and incremental synchronization of suppliers' information between PeopleSoft Payables and Oracle RMS and discusses the following topics:

- [Section 2.3.2, "Supplier Integration Details"](#)
- [Section 2.3.3, "PeopleSoft Payables Interfaces"](#)
- [Section 2.3.4, "Oracle RMS Interfaces"](#)
- [Section 2.3.5, "Core AIA Components"](#)
- [Section 2.3.6, "Integration Services"](#)

2.3.1 Supplier Integration Overview

In the integrated environment, PeopleSoft Payables acts as a payable and accounting engine, and Oracle RMS handles supplier payments, merchandise write-offs, and prepaid adjustments.

Merchandise suppliers supply goods and services that a retailer sells to customers. PeopleSoft Payables and Oracle RMS must share suppliers' information. Oracle RMS requires the supplier information for several key functions, including creation and management of items and purchase orders. PeopleSoft Payables requires the supplier information for supplier payment. For end-to-end business integration, the two systems must share the same supplier instance and related information.

PeopleSoft Payables is the source of valid suppliers (vendors in PeopleSoft Payables) and their Remit to Location and Order from addresses, and the relationships between this supplier data and the financial business unit structure.

The supplier integration synchronizes supplier information from PeopleSoft Payables to Oracle RMS through these integration flows:

- Load Initial Suppliers from PeopleSoft Payables to Oracle RMS. Enables the loading of all active merchandise suppliers, the current effective supplier locations and their current effective remit and order-to addresses, and the relationship between the supplier data and the financial business unit structure from PeopleSoft Payables to Oracle RMS for a new instance (logical or physical) of Oracle RMS. It is a pre-implementation stage. PeopleSoft suppliers must be classified as supplier and have the *open for ordering* attribute for the initial load.
- Incremental Creation and Updates of Suppliers from PeopleSoft Payables to Oracle RMS. Enables the synchronization of incremental creation and updates of the current effective-dated suppliers from PeopleSoft Payables to Oracle RMS.

For more information about supplier create and update operations, see *PeopleSoft Enterprise Source to Settle Common Information PeopleBook*, "Maintaining Vendor Information."

This integration is not a point-to-point integration between PeopleSoft Payables and Oracle RMS. An Oracle AIA layer serves as an intermediate thin layer of application between PeopleSoft Payables and Oracle RMS. As part of the supplier integration, PeopleSoft Payables sends supplier information to the Oracle AIA layer, and the Oracle AIA layer delivers the information to Oracle RMS. The Oracle AIA layer performs message filtering, message transformation, and message routing. Because this integration is point-to-point, the vendor number (ID) in PeopleSoft is not similar to the supplier number (ID) in Oracle Retail.

2.3.1.1 Solution Assumptions and Constraints

The integration design assumes that:

1. PeopleSoft Payables is the source system for merchandise suppliers, their contacts, locations, addresses, and business unit structure.
2. You can create suppliers and suppliers' locations in PeopleSoft Payables.
You can maintain the relationship between suppliers, suppliers' locations, and financial structure in PeopleSoft Payables. This integration is a one-way synchronization. Any update to supplier information in Oracle RMS is not synchronized with PeopleSoft Payables.
3. The volume of payload (SupplierPartyList) that the process handles depends on the server configuration.
PeopleSoft sends the suppliers in batches based on different criteria.
4. PeopleSoft Payables sends all current effective addresses in the message.
This message contains, for each trading location, specific and detailed Order from and Remit to addresses.
5. The PeopleSoft system sends all related information to Oracle AIA.
For example, if an address changes, then all the current effective locations with that address and all the suppliers linked to that location are included in the message. The same is true for the reverse: if supplier information changes, then all its locations and addresses are sent.
6. The PeopleSoft system sends the COMMON field for the DVM lookup.
DVM lookup in the PeopleSoft ABCS is unnecessary because the DVM lookup is done only in the Oracle RMS side Provider ABCS.
7. PeopleSoft suppliers have a record in the Oracle Retail language before the performance of the sync operation.

The last update to a supplier in the PeopleSoft application should be in the Oracle Retail primary language, so that the translatable fields appear correct in Oracle Retail.

8. PeopleSoft Payables sends supplier information for approved, unapproved, inactive, and archived suppliers.

The Oracle AIA layer for the Retail side filters and allows only approved and inactive suppliers during the create operation. If a supplier is active but not open for ordering, the Oracle AIA layer changes its status to inactive. For the update operation, the Oracle AIA layer allows only approved suppliers.

These are the constraints for the suppliers' integration:

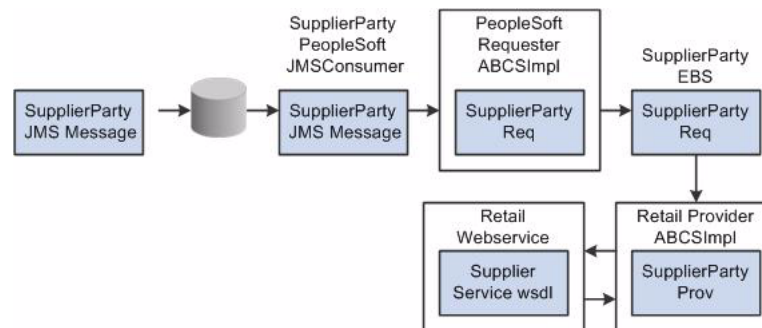
1. The Oracle AIA services do not know the changes in the supplier unless PeopleSoft Payables notifies them.
2. The Oracle AIA integration solutions do not correct failure in the processing of messages in Oracle RMS. Oracle AIA only sends a fault message.

PeopleSoft Payables does not automatically send the same message again. In some cases, if a message fails on the PeopleSoft side, you use a manual process to resubmit the message.

3. PIP is responsible only for invoking the Oracle RMS web service.

Table 2–19 illustrates the supplier integration flow.

Figure 2–19 Supplier integration flow



2.3.2 Supplier Integration Details

These services are common to *Create*, *Update*, and *Sync* supplier integration flows:

- SupplierPartyPeopleSoftJMSConsumer
- SupplierPartyEBS

These services are specific to *Sync* supplier integration flows:

- SyncSupplierPartyListPeopleSoftReqABCSImpl
- SyncSupplierPartyListRetailProvABCSImpl

These services are specific to *Create* supplier integration flows:

- CreateSupplierPartyPeopleSoftReqABCSImpl
- CreateSupplierPartyRetailProvABCSImpl

These services are specific to *Update* supplier integration flows.

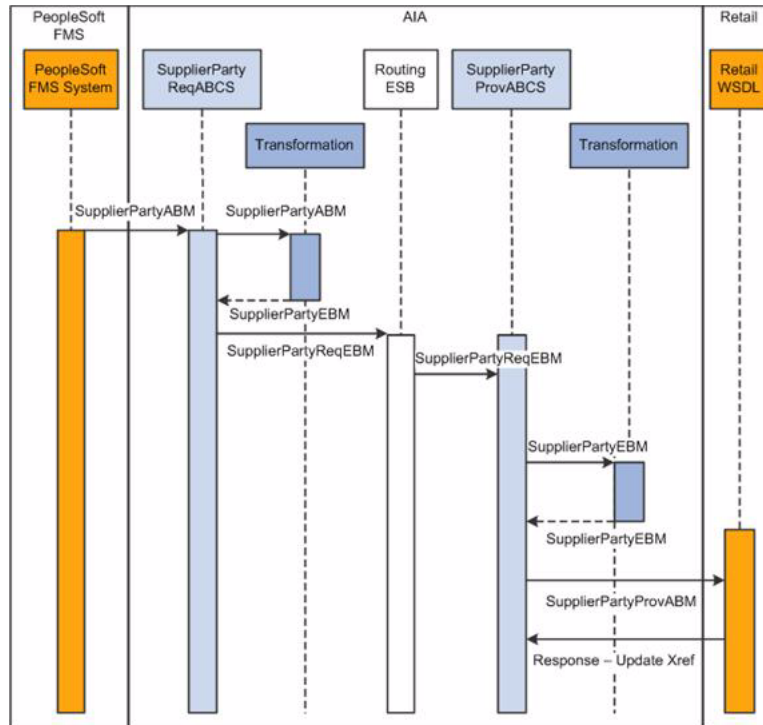
- UpdateSupplierPartyRetailProvABCSImpl

- UpdateSupplierPartyPeopleSoftReqABCImpl

2.3.2.1 Sequence Diagram

Table 2–20 illustrates the supplier information integration flow.

Figure 2–20 Supplier integration sequence diagram



This sequence diagram is applicable to initial load, create, and update integration flows.

When you initiate the process:

1. PeopleSoft Payables puts the SyncSupplierPartyListPSFTABM, CreateSupplierPartyPSFTABM, and UpdateSupplierPartyPSFTABM in the AIA_PeopleSoftSupplierPartyJMSQueue with MessageName set to SyncSupplierPartyList/CreateSupplierParty/UpdateSupplierParty.
2. The SupplierPartyPeopleSoftJMSConsumer service listens on the AIA_PeopleSoftSupplierPartyJMSQueue and invokes SyncSupplierPartyListPeopleSoftReqABCImpl for SyncSupplierPartyListPSFTABM, CreateSupplierPartyPeopleSoftReqABCImpl for CreateSupplierPartyPSFTABM and UpdateSupplierPartyPeopleSoftReqABCImpl for UpdateSupplierPartyPSFTABM.

3. The SyncSupplierPartyListPeopleSoftReqABCImpl transforms the SupplierPartyListABM message into SupplierPartyListEBM.

Transformation does cross-referencing for system-specific values and calls the SupplierPartyEBS with operation SyncSupplierPartList. SupplierPartyEBS is a Mediator service with several operations on the supplier EBO.

4. The CreateSupplierPartyPeopleSoftReqABCImpl transforms the SupplierPartyABM message into SupplierPartyEBM.

Transformation does cross-referencing for system-specific values and calls the SupplierPartyEBS with operation CreateSupplierParty. SupplierPartyEBS is a Mediator service with several operations on the supplier EBO.

5. The UpdateSupplierPartyPeopleSoftReqABCImpl transforms the SupplierPartyABM message into SupplierPartyEBM.

Transformation does cross-referencing for system-specific values and calls the SupplierPartyEBS with operation UpdateSupplierParty. SupplierPartyEBS is a Mediator service with several operations on the supplier EBO.

6. SupplierPartyEBS routes the SupplierPartyListEBM to Retail provider ABCS implementation. It also provides CAVS routing.
7. SyncSupplierPartyListRetailProvABCImpl transforms SupplierPartyListEBM to Retail supplier message ABM.

Transformation applies the DVM and invokes create or update web service from Oracle RMS. It also updates the cross-reference table after the Retail web service call.

8. CreateSupplierPartyRetailProvABCImpl transforms SupplierPartyEBM to Retail supplier message ABM.

Transformation applies the DVM and invokes a create web service from Oracle RMS. It also updates the cross-reference table after the Retail web service call.

9. UpdateSupplierPartyRetailProvABCImpl transforms SupplierPartyEBM to Retail supplier message ABM.

Transformation applies the DVM and invokes an update web service from Oracle RMS. It also updates the cross-reference table after the Retail web service call.

Note: The structure of the PeopleSoft ABM is very close to that of an EBM. PeopleSoft ABM does not have some details, such as the EBM header and the Oracle AIA namespace. These details are added by the PeopleSoft ABCS. So the PeopleSoft objects are referred as ABMs even though they have the content structured in the EBM schema.

2.3.2.2 Multiple Language Support

Oracle Retail supports multiple languages, however, one language is considered primary.

The PeopleSoft system supports several languages per instance based on user localization preferences. Any translatable data attribute (such as supplier name) may possibly have multiple occurrences, each one representing the language of the user who entered or updated it in the PeopleSoft system. For example, the data field for the supplier name has a French entry from a user with a French UI preference and an English entry from a user with an English UI preference.

Because Oracle Retail supports only one primary language per instance, the Oracle AIA layer filters Oracle Retail records on the provider based on the EBM header language code. The primary language supported by the Oracle Retail instance is identified in AIAConfigurationProperties.XML.

The filtering operation includes a DVM language code lookup based on the common EBM Header. If the DVM language code lookup returns the same value stored in the AIAConfigurationProperties.XML, then the Oracle Retail record on the provider is accepted.

Constraints

For the incremental creates and updates of a supplier, Oracle AIA does not filter the message from PeopleSoft. Oracle AIA sends the message in the same language in which it was created or updated, even if this language is not the primary language of the Oracle Retail instance.

Oracle AIA on the Retail side filters initial load, create, and update messages and sends only those messages that have a language code similar to the Oracle Retail primary language code.

2.3.3 PeopleSoft Payables Interfaces**Outbound Interactions**

- Sync SupplierParty (Asynchronous, one-way)
Request Schema: AP_VENDOR_SYNC_EBM.V1.xsd
- Create SupplierParty (Asynchronous, one-way)
Request Schema: AP_CREATE_VENDOR_SYNC_EBM.V1.xsd
- Update SupplierParty (Asynchronous, one-way)
Request Schema: AP_UPDATE_VENDOR_SYNC_EBM.V1.xsd

2.3.4 Oracle RMS Interfaces**Inbound Interactions**SupplierServicePortFactory

Operation: createSupplierCollectionDesc, Request Schema: SupplierCollectionDesc.xsd, Response Schema: SupplierCollectionRef.xsd

Operation: updateSupplierCollectionDesc, Request Schema: SupplierCollectionDesc.xsd, Response Schema: SupplierCollectionRef.xsd

Operation: createSupplierDesc, Request Schema: SupplierDesc.xsd, Response Schema: SupplierRef.xsd

Operation: updateSupplierDesc, Request Schema: SupplierDesc.xsd, Response Schema: SupplierRef.xsd

For more information, see the *RMS Operations Guide*, "Web Services Overview."

2.3.5 Core AIA Components

The supplier Information integration uses these components:

- Merchandise Supplier EBO (SupplierPartyEBO)
- Merchandise Supplier EBS (SupplierPartyEBS)
- Merchandise Supplier EBM (SupplierPartyEBM)

The core EBO and EBM XSD files can be located by EBO within this parent folder:
\$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core EBS WSDL files can be located by EBO within this parent folder: \$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation about individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in Oracle Enterprise Repository.

For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Extensibility for Oracle AIA Artifacts," Extending EBOs.

2.3.6 Integration Services

These services are delivered with the process integration for supplier information:

- SupplierPartyPeopleSoftJMSConsumer
- SyncSupplierPartyListPeopleSoftReqABCImpl
- CreateSupplierPartyPeopleSoftReqABCImpl
- UpdateSupplierPartyPeopleSoftReqABCImpl
- SupplierPartyEBS
- SyncSupplierPartyListRetailProvABCImpl
- CreateSupplierPartyRetailProvABCImpl
- UpdateSupplierPartyRetailProvABCImpl

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

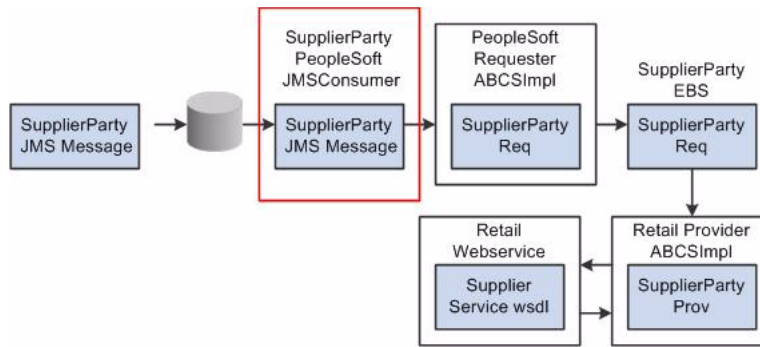
2.3.6.1 SupplierPartyPeopleSoftJMSConsumer

SupplierPartyPeopleSoftJMSConsumer is a Mediator service. It has three JMSAdapter services for dequeue of the message from the queue. After dequeuing the messages, it calls for the respective PeopleSoftReqABCImpl (*Create*, *Update*, or *Sync*).

This service has these JMS adapters:

- SyncSupplierPartyPeopleSoftJMSConsumer
- CreateSupplierPartyPeopleSoftJMSConsumer
- UpdateSupplierPartyPeopleSoftJMSConsumer

[Table 2–21](#) illustrates the relationship of the SupplierPartyPeopleSoftJMSConsumer with the other services in the integration flow.

Figure 2–21 Relationship of SupplierPartyPeopleSoftJMSConsumer to other services

2.3.6.2 SyncSupplierPartyListPeopleSoftReqABCImpl

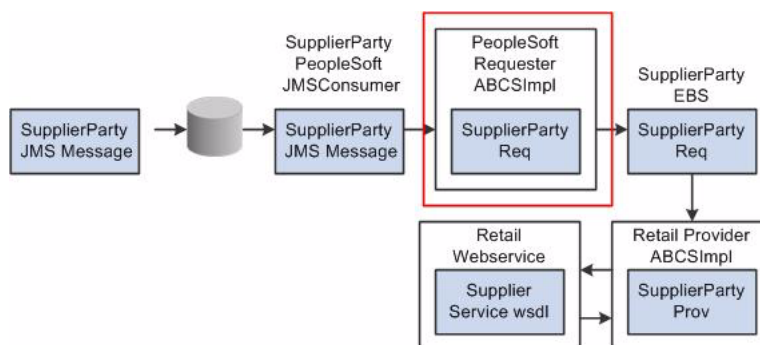
SupplierPartyPeopleSoftJMSConsumer invokes the SyncSupplierPartyListPeopleSoftReqABCImpl service for the list of existing suppliers during the initial integration with other applications.

This service is a single operation service that has SupplierPartyEBS as a partner service. It accepts a PeopleSoft SupplierPartyListEBM message as a request and does not return a response to the calling service. This service is a BPEL process.

This service performs these actions:

- Accepts a SyncSupplierPartyListReqMsg message from PeopleSoft. This message contains a cross-reference for suppliers and suppliers' addresses.
- Transforms SyncSupplierPartyListReqMsg to SyncSupplierPartyListEBM. While it is transforming ABM to EBM, it looks up cross-references for SUPPLIERPARTY_ID, SUPPLIERPARTY_LOCATION_ID, and SUPPLIERPARTY_ADDRESS_ID.
- Sends SyncSupplierPartyListEBM messages as input to the SyncSupplierPartyList operation in the SupplierPartyEBS service.

Table 2–22 illustrates the relationship of the SyncSupplierPartyListPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–22 Relationship of SyncSupplierPartyListPeopleSoftReqABCImpl to other services

2.3.6.3 CreateSupplierPartyPeopleSoftReqABCImpl

This service is a single operation service that has SupplierPartyEBS as a partner service. This service is invoked by SupplierPartyPeopleSoftJMSConsumer as a part of the new supplier information synchronization. This service is a BPEL process.

This service performs these actions:

- Accepts a CreateSupplierPartyReqMsg as a request and does not return a response to the calling service.

This message contains a cross-reference for suppliers and suppliers' addresses.

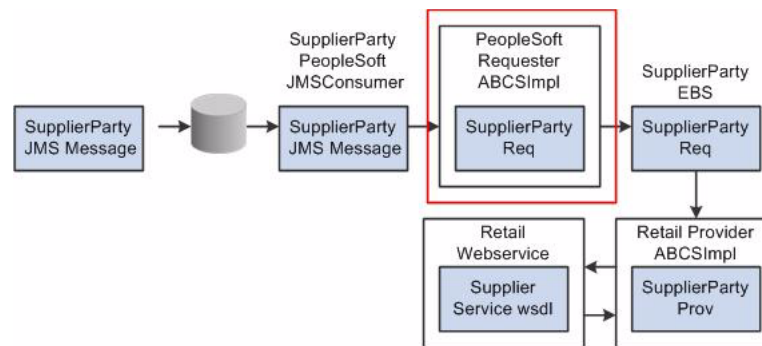
- Transforms CreateSupplierPartyReqMsg to CreateSupplierPartyEBM.

While it is transforming ABM to EBM, it looks up cross-references for SUPPLIERPARTY_ID, SUPPLIERPARTY_LOCATION_ID, and SUPPLIERPARTY_ADDRESS_ID.

- Sends the CreateSupplierPartyEBM message as an input to the CreateSupplierParty operation in the SupplierPartyEBS service.

Table 2–23 illustrates the relationship of the CreateSupplierPartyPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–23 Relationship of CreateSupplierPartyPeopleSoftReqABCImpl to other services



2.3.6.4 UpdateSupplierPartyPeopleSoftReqABCImpl

This service is a single operation service that has SupplierPartyEBS as a partner service. It accepts a PSFT supplier message as a request and does not return a response to the calling service. This service is a BPEL process triggered by SupplierPartyPeopleSoftJMSCConsumer.

The process performs these actions:

- Accepts the UpdateSupplierPartyReqMsg message from PeopleSoft Payables.

This message contains a cross-reference for suppliers and suppliers' addresses.

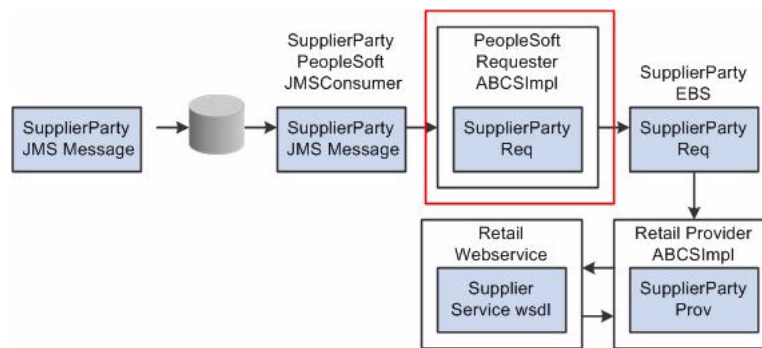
- Transforms UpdateSupplierPartyReqMsg to UpdateSupplierPartyEBM.

While it is transforming ABM to EBM, it looks up a cross-reference for SUPPLIERPARTY_ID, SUPPLIERPARTY_LOCATION_ID, and SUPPLIERPARTY_ADDRESS_ID.

- Sends the UpdateSupplierPartyEBM message as an input to the UpdateSupplierParty operation in the SupplierPartyEBS service.

Table 2–24 illustrates the relationship of the UpdateSupplierPartyPeopleSoftReqABCImpl with the other services in the integration flow.

Figure 2–24 Relationship of UpdateSupplierPartyPeopleSoftReqABCImpl to other services

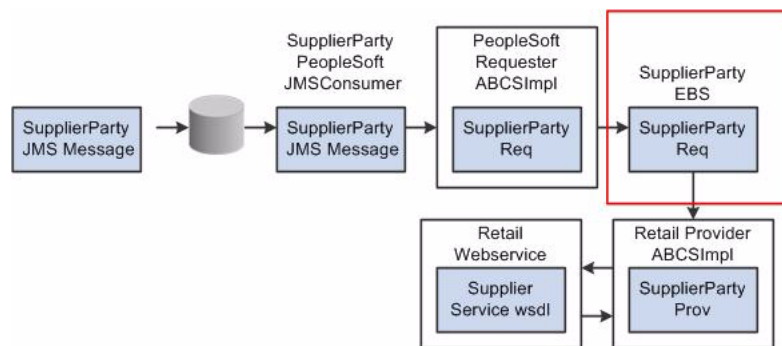


2.3.6.5 SupplierPartyEBS

SupplierPartyEBS performs all supplier-related actions, such as *Create Supplier*, *Update Supplier*, and *Sync Supplier*. It routes incoming *Sync*, *Update*, or *Create* messages to the respective ProviderABCImpl. No transformation is done in this EBS.

Table 2–25 illustrates the relationship of the SupplierPartyEBS with the other services in the integration flow.

Figure 2–25 Relationship of SupplierPartyEBS to other services



These operations on the SupplierPartyEBS are used in the supplier information integration flow:

- SyncSupplierPartyList
- CreateSupplierParty
- UpdateSupplierParty

For more information about this EBS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Developing EBSs" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding EBSs."

2.3.6.6 SyncSupplierPartyListRetailProvABCImpl

SyncSupplierPartyListRetailProvABCImpl is a single operation service. This service is a BPEL process.

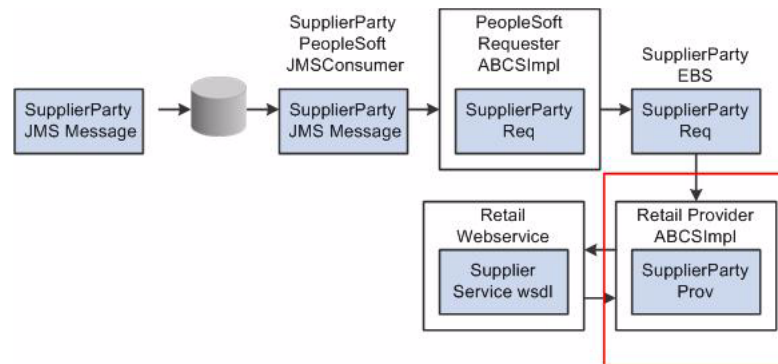
This process performs these actions:

- Receives input from SupplierPartyEBS.

- Accepts a SyncSupplierPartyListEBM message. A SyncSupplierPartyListEBM message contains these details:
 - Standard supplier attributes
 - Supplier contacts information
 - Supplier location information
 - Supplier location contact information
 - Supplier location remit address information
 - Supplier location remit address contact information
 - Supplier location order address information
 - Supplier location order address contact information
 - Supplier location business unit
- Transforms SyncSupplierPartyListEBM into a SyncSupplierRetailABM message. During transformation, the DVM lookups change the value for language code, state, country code, currency code, SupplierParty address type, and SupplierParty status code.
- Invokes the Create/UpdateSupplierPartyList WSDL after transforming the message from the SyncSupplierEBS to SyncSupplierPartyList ABM and wrapping the ABM in WSDL.
- Updates the cross-references for Oracle Retail after the Oracle Retail web service call.

Table 2–26 illustrates the relationship of the SyncSupplierPartyListRetailProvABCImpl with the other services in the integration flow.

Figure 2–26 Relationship of SyncSupplierPartyListRetailProvABCImpl to other services



2.3.6.7 CreateSupplierPartyRetailProvABCImpl

CreateSupplierPartyRetailProvABCImpl is a single operation service. It is a BPEL process.

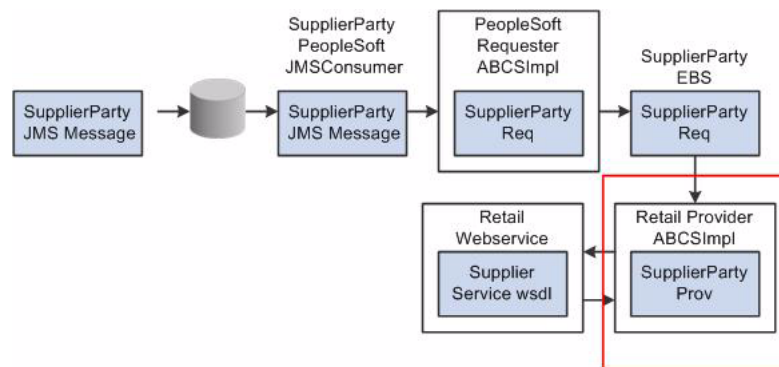
This process performs these actions:

- Receives input from SupplierPartyEBS.

- Invokes the RMS web service after transforming the message from the SupplierPartyEBS to CreateSupplierParty ABM and wrapping the ABM in a WSDL message.
- Accepts a CreateSupplierPartyEBM message. A CreateSupplierPartyEBM message contains these details:
 - Standard supplier attributes
 - Supplier contacts information
 - Supplier location information
 - Supplier location contact information
 - Supplier location remit address information
 - Supplier location remit address contact information
 - Supplier location order address information
 - Supplier location order address contact information
 - Supplier location business unit
- Transform CreateSupplierPartyEBM into a CreateSupplierRetailABM message. During transformation, the DVM lookups change the value for language code, state, country code, currency code, SupplierParty address type, and SupplierParty status code. After CreateSupplierRetailABM message construction, the Oracle RMS web service is invoked for create.
- Updates the cross-references for Oracle Retail after the Oracle Retail web service call.

Table 2–27 illustrates the relationship of the CreateSupplierPartyRetailProvABCImpl with the other services in the integration flow.

Figure 2–27 Relationship of CreateSupplierPartyRetailProvABCImpl to other services



2.3.6.8 UpdateSupplierPartyRetailProvABCImpl

UpdateSupplierPartyRetailProvABCImpl is a single operation service that receives input from SupplierPartyEBS. It invokes the Update web service in Oracle RMS after transforming the message from the SupplierPartyEBS to UpdateSupplierParty ABM and wrapping the ABM in a WSDL message. It is a BPEL process.

The process performs these actions:

- Accepts an UpdateSupplierPartyEBM message.
 - An UpdateSupplierPartyEBM message contains these details:

- Standard supplier attributes
- Supplier contacts information
- Supplier location information
- Supplier location contact information
- Supplier location remit address information
- Supplier location remit address contact information
- Supplier location order address information
- Supplier location order address contact information
- Supplier location business unit
- Transforms UpdateSupplierPartyEBM into UpdateSupplierRetailABM.
During transformation, the DVM lookups change the value for language code, state, country code, currency code, SupplierParty address type, and SupplierParty status code.
- Updates the cross-references for Oracle Retail after the Oracle Retail web service call.

Table 2–28 illustrates the relationship of the UpdateSupplierPartyRetailProvABCImpl with the other services in the integration flow.

Figure 2–28 Relationship of UpdateSupplierPartyRetailProvABCImpl to other services

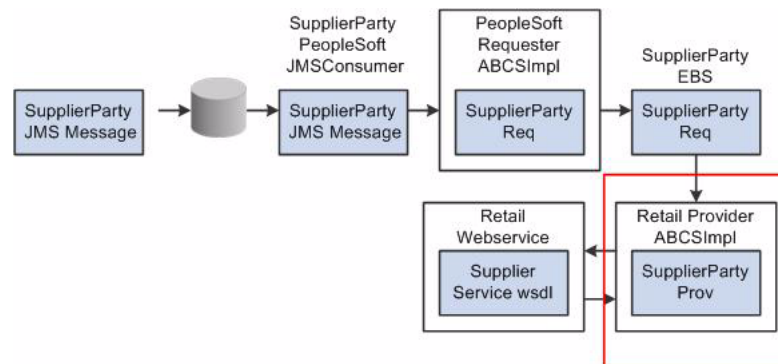


Chart of Accounts Validation Integration Flow

This chapter describes how to request a chart of accounts combination validation and discusses Oracle Retail and PeopleSoft General Ledger (GL) interfaces, core Oracle Application Integration Architecture (Oracle AIA) components, and integration services.

This chapter includes the following sections:

- [Section 3.1, "Requesting Chart of Accounts Combination Validation"](#)
- [Section 3.2, "Oracle Retail Interfaces"](#)
- [Section 3.3, "PeopleSoft GL Interfaces"](#)
- [Section 3.4, "Core AIA Components"](#)
- [Section 3.5, "Integration Services"](#)

3.1 Requesting Chart of Accounts Combination Validation

PeopleSoft GL is the system of record for chart of accounts segment combinations. You set up chart of accounts ChartFields (segments) combinations manually in the Oracle Retail Merchandising System (Oracle RMS), Oracle Retail Invoice Matching (Oracle ReIM), and Oracle Retail Sales Audit (Oracle ReSA). This manual setup enables you to assign transaction data in sales audit, stock ledger, and invoice match to specific account codes. Oracle ReIM can also create segment combinations dynamically during the invoice match transaction processing.

While creating the valid segment combinations, Oracle Retail validates each created combination individually with PeopleSoft GL. Any valid segment combination in Oracle Retail must exist as a valid combination in PeopleSoft GL. Oracle Retail publishes the segment combination to PeopleSoft GL for validation. PeopleSoft GL verifies the combination and returns the status to Oracle Retail. If the combination is valid, the combination is stored in the Oracle Retail database tables.

Oracle Retail sends these data for validation to PeopleSoft GL:

- Requesting system (Oracle RMS, Oracle ReIM, or Oracle ReSA)
- Set of books (GL business unit)
- Segment or ChartField combination values

PeopleSoft GL sends the validation status whether the combination is valid or invalid. The Oracle AIA layer copies all the other information from the original request and sends these response data to Oracle Retail:

- Requesting system (Oracle RMS, Oracle ReIM, or Oracle ReSA)
- Set of books (GL business unit)
- Valid segment or ChartField combination values
- Validation status - valid or invalid

3.1.1 Solution Assumptions and Constraints

This design assumes that:

1. The Oracle Retail request application business message (ABM) contains a field called `requesting_system` that identifies the system that is requesting the service so that the correct Retail segment is retrieved from the Domain Value Map (DVM) lookup.

DVM values (segments to PeopleSoft accounts) may be set up differently for each system (Oracle RMS, Oracle ReSA, or Oracle ReIM).
2. PeopleSoft GL accepts all the Oracle AIA Common values in the provider ABM.

Therefore, no DVM lookups are required on the PeopleSoft provider side. The DVM values are entered in Oracle AIA and PeopleSoft.
3. Multiple sets of GL accounts can be sent from Oracle Retail to the PeopleSoft validation web service.

3.1.2 Chart of Accounts Combination Validation Integration Flow

These services are delivered with the chart of accounts combination validation integration flow:

- `ProcessGLAccountValidationRetailReqABCSImpl`
- `ChartOfAccountsEBS`
- `ProcessGLAccountValidationPeopleSoftProvABCSImpl`

[Figure 3-1](#) illustrates the chart of accounts combination validation integration flow.

3.2 Oracle Retail Interfaces

Retail Outbound Interactions

- Name: ProcessGLAccountValidation: Oracle Retail (Oracle RMS, Oracle RESA, or Oracle REIM) invokes this service to determine whether an account is valid with PeopleSoft GL.
- Interaction Pattern: Synchronous Request - Response
- Request Schema: GLAcctCollectionDesc.xsd, GLAcctDesc.xsd
- Response Schema: GLAcctCollectionRef.xsd, GLAcctRef.xsd.

3.3 PeopleSoft GL Interfaces

PeopleSoft Inbound Interactions

- Name: GL_CHARTFIELD.1 (The ProcessGLAccountValidation request invokes this web service. It sends a response indicating whether the account is valid. The request and response schema for PeopleSoft are in the format of the ProcessGLAccountAccountValidationEBM element of the existing ChartOfAccountsEBM.xsd.
- Interaction Pattern: Synchronous Request-Response
- Request Schema: ChartFieldComboEditReq.V1.xsd
- Response Schema: ChartFieldComboEditRes.V1.xsd

For more information, see *PeopleSoft Enterprise General Ledger PeopleBook*, "Integrating and Transferring Information Among Applications."

3.4 Core AIA Components

The chart of accounts combination validation integration flow uses these components:

- ChartOfAccountsEBO
- ChartOfAccountsEBM
- ChartOfAccountsEBS

The core enterprise business object (EBO) and enterprise business message (EBM) XSD files can be located by EBO within this parent folder: \$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core enterprise business service (EBS) WSDL files can be located by EBO within this parent folder: \$AIA_HOME/AIAMetaData/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation of individual EBOs and EBMs, click the AIA Reference Doc link on EBO and EBM detail pages in Oracle Enterprise Repository.

For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

EBOs can be extended, for instance, to add new data elements. These extensions are protected and remain intact after a patch or an upgrade.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Extensibility for Oracle AIA Artifacts," Extending EBOs.

3.5 Integration Services

The chart of accounts combination validation integration flow uses these services:

- ProcessGLAccountValidationRetailReqABCSImpl
- ChartOfAccountsEBS
- ProcessGLAccountValidationPeopleSoftProvABCSImpl

You can use the Integration Scenario Summary page in the Oracle AIA Console to search for and view integration scenarios that use a particular Applications Business Connector Service (ABCS).

For more information about using the Oracle Enterprise Repository and configuring it to provide the AIA Reference Doc link, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

3.5.1 ProcessGLAccountValidationRetailReqABCSImpl

Oracle Retail (Oracle RMS, Oracle ReSA, or Oracle ReIM) invokes this requester ABCS to validate a GL account. This service takes ProcessGLAccountValidationRetailABM (GLAcctDesc.xsd) to validate a GL account. It transforms ProcessGLAccountValidationRetailABM into the ProcessGLAccountValidationEBM and then invokes ChartOfAccountsEBS. When ChartOfAccountsEBS returns with ProcessGLAccountValidationResponseEBM, it is then transformed into ProcessGLAccountValidationResponseRetailABM (GLAcctRef.xsd) and sent to the calling application.

In this requester ABCS, the Set of Books ID is mapped to the ChartOfAccountIdentification field in the EBM. During this transformation, the Set of Books ID uses the BUSINESS_UNIT DVM lookup to transform the Set of Books ID from Oracle Retail, and retrieves the common value and loads it into the ChartOfAccountIdentification field in the EBM.

The segments (1-20) are mapped to the GLElementCode field in the EBM. During this transformation, the segments (1-20) use the CHARTOFACTS_ACCOUNTS_GLELEMENT DVM lookup to:

- Transform the attributes from Oracle Retail.
- Retrieve the common value and load it into the GLElementCode field in the EBM.

When ProcessGLAccountValidationResponseEBM returns, regardless of whether the account is valid, a transformation occurs to populate the GLAcctRef account_status field. A CHARTOFACTS_ACCOUNTS_ACCOUNTSTATUS DVM table is used to transform the Oracle AIA Common/PSFT value to the Oracle Retail value.

ProcessGLAccountValidationRetailReqABCSImplExt is a partner link service. It is an extensibility service that takes ProcessGLAccountValidationRetailABM, ProcessGLAccountValidationRetailABM, ProcessGLAccountValidationEBM, and ProcessGLAccountValidationResponseEBM as inputs and returns the same. This service enables the customer to filter, augment, and validate the input or output. In this same transformation, you must populate the retail set of books id and segment values in the ProcessGLAccountValidationResponseRetailABM (GLAcctRef) with the

same values that were passed in the ProcessGLAccountValidationRetailABM (GLAcctDesc).

3.5.2 ChartOfAccountsEBS

ChartOfAccountsEBS calls ProcessGLAccountValidationRetailReqABCImpl. It expects the ProcessGLAccountValidationEBM encapsulating the ChartOfAccountsEBO and sends the ProcessGLAccountValidationEBM to the PeopleSoft ChartField validation service. In turn, it returns whether the account is valid to the Retail ProcessGLAccountValidationRetailReqABCImpl service.

For more information about this EBS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Designing and Developing EBSs" and *Oracle Fusion Middleware Concepts and Technologies Guide for Oracle Application Integration Architecture Foundation Pack*, "Understanding EBSs."

3.5.3 ProcessGLAccountValidationPeopleSoftProvABCImpl

The ProcessGLAccountValidationPeopleSoftProvABCImpl service receives ProcessGLAccountValidationEBM as input from ChartOfAccountsEBS. This service is a single operation service. This service expects ProcessGLAccountValidationEBM and publishes ProcessGLAccountValidationEBM to PeopleSoft Code Combination Validate application programming interface (API). In turn, it returns a notification of validation results to ChartOfAccountsEBS.

ProcessGLAccountValidationEBM is transformed into a ChartFieldComboEditReq.V1 ABM. This transformed object is sent to the GL_CHARTFIELD PeopleSoft service for validation of chart of accounts. Because PeopleSoft accepts the EBM itself, no DVM lookups are required. All the values passed into the service are in the Oracle AIA common format.

When the ChartOfAccountsEBS service invokes the ProcessGLAccountValidationPeopleSoftProvABCImpl service:

- ChartOfAccountsEBS calls ProcessGLAccountValidationPeopleSoftProvABCImpl with ProcessGLAccountValidationEBM. EBM_Header variable is assigned with the header information of ProcessGLAccountValidationEBM.
- The ProcessGLAccountValidationPeopleSoftProvABCImplExt PreProcessEBM is called with ProcessGLAccountValidationEBM based on a parameter in the AIAConfig file that is for custom extensibility.
- ProcessGLAccountValidationEBM (ChartOfAccountsEBM.xsd) transforms into ProcessGLAccountValidationEBM (ChartFieldComboEditReq.V1.xsd). This transformation does not apply the BUSINESS_UNIT and CHARTOFACCOUNTS_GLELEMENT DVM lookups. It accepts the Oracle AIA common values.
- ProcessGLAccountValidationPeopleSoftProvABCImplExt PreProcessABM is called with ProcessGLAccountValidationEBM based on a parameter in the AIAConfig file.
- ProcessGLAccountValidationPeopleSoftProvABCImplExt PostProcessABM is called with ProcessGLAccountValidationResponseEBM based on a parameter in the AIAConfig file.
- ProcessGLAccountValidationResponseEBM transforms into ProcessGLAccountValidationResponseEBM. DVM lookups are not required for this transformation.

- ProcessGLAccountValidationPeopleSoftProvABCSExt PostProcessEBM is called with ProcessGLAccountValidationResponseEBM based on a parameter in the AIAConfig file.
- ProcessGLAccountValidationPeopleSoftProvABCSExt returns ProcessGLAccountValidationResponseEBM to the caller ChartOfAccountsEBS.

ProcessGLAccountValidationPeopleSoftProvABCSExt is a partner link service. This service is an extensibility service that takes ProcessGLAccountValidationEBM/ProcessGLAccountValidationResponseEBM as input and returns the same. This service enables the user to filter, augment, and validate the input or output.

Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries

This chapter provides an overview of the process integration for accounting entries from Oracle Retail Sales Audit (Oracle ReSA), Oracle Retail Merchandising System (Oracle RMS), and Oracle Retail Invoice Matching (Oracle ReIM) to PeopleSoft General Ledger (GL). It discusses how to perform setup tasks, configure and generate data in Oracle Retail, pick up and transform the data, and configure and run the process integration for accounting entries.

This chapter includes the following sections:

- [Section 4.1, "Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview"](#)
- [Section 4.2, "Performing Setup Tasks"](#)
- [Section 4.3, "Configuring and Generating Data in Oracle Retail"](#)
- [Section 4.4, "Picking Up and Transforming the Data"](#)
- [Section 4.5, "Configuring and Running the Process Integration for Accounting Entries"](#)

4.1 Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview

The process integration for accounting entries enables you to record the financial effect of changes to sellable store and warehouse inventory. It also records the financial effect of sales and returns, cash reconciliation, and void transactions from stores.

The system sends sales audit and stock ledger data in Oracle Retail to PeopleSoft General Ledger (GL) through accounting entries. Similarly, the system sends the prepaid invoice reversals and write-offs of aged receipts to PeopleSoft GL. Oracle Retail Merchandising System (Oracle RMS), Oracle Retail Invoice Matching (Oracle ReIM), and Oracle Retail Sales Audit (Oracle ReSA) are the sources for accounting entry. After successful completion of data transfer, the system deletes the data from the source interface tables.

This integration uses the Oracle Data Integrator (ODI) application to transfer the data from the Oracle Retail applications to the PeopleSoft GL application.

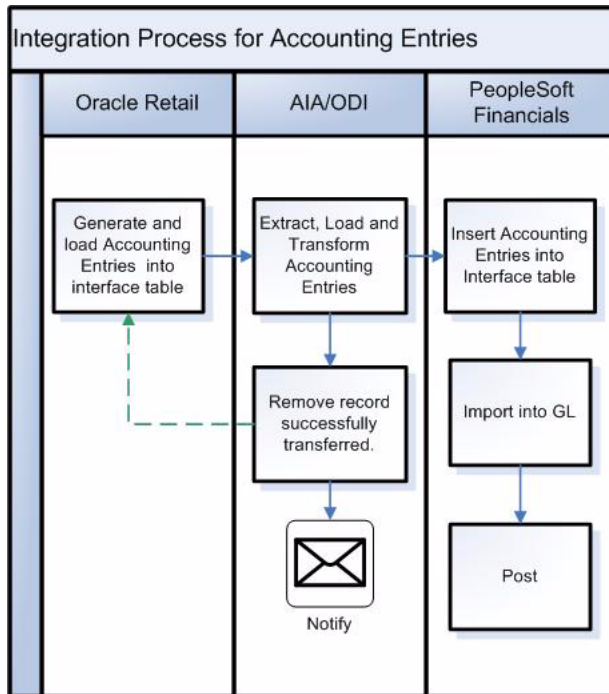
The process integration between Oracle Retail (Oracle RMS, Oracle ReSA, and Oracle ReIM) and PeopleSoft GL supports these integration flows:

- Post stock ledger from Oracle RMS to PeopleSoft GL.
- Post channel sales, cash, and deposits from Oracle ReSA to PeopleSoft GL.
- Post prepaid invoice reversals after receipt from Oracle ReIM to PeopleSoft GL.

4.1.1 Business Process Flow for the Integration of Accounting Entries

Figure 4–1 illustrates the process integration for accounting entries.

Figure 4–1 Business process flow for the integration of accounting entries



This table describes the overall process:

Table 4–1 Integration of Accounting Entries Overall Process

Work Location	Step
Oracle Retail	1. Configure, generate, and load accounting entries into interface table in Oracle Retail.
Integration process	2. The integration process waits for the data to arrive in the Oracle Retail interface (staging) table. 3. Load the Oracle Retail data into PeopleSoft GL and transform it. 4. Delete the data from the interface table after a successful load.
PeopleSoft GL	5. Import from accounting entries interface table to journal entries and then post to the GL accounts.

4.1.2 Facts and Constraints

Oracle Retail

1. Oracle RMS, Oracle ReSA, and Oracle ReIM are the source of the accounting entry flow.
2. Oracle Retail populates a staging table, STG_FIF_GL_DATA or IM_FINANCIALS_STAGE (depending on whether the Oracle RMS or Oracle ReIM package is running) on a scheduled basis.
3. Data in the Oracle Retail staging tables is deleted after the ODI job completes successfully.

PeopleSoft GL

1. PeopleSoft GL has one table to receive data: PS_ORT_ACCT_ENTRY.
2. The PeopleSoft Journal Generator process is run using the table to create the accounting entries in PeopleSoft GL.
3. PeopleSoft GL supports both databases; however, this process integration pack (PIP) only supports Oracle and IBM UDB DB2 databases.
4. Data in the PeopleSoft table is flagged as processed by PeopleSoft GL.

Constraints

The monetary amount fields in the PeopleSoft GL and Oracle Retail systems have a mismatch in the number of digits after the decimal. The PeopleSoft system allows 3 digits after the decimal, and Oracle Retail allows 4 digits. Therefore, some data may be rounded while being transferred from Oracle Retail to PeopleSoft GL.

4.2 Performing Setup Tasks

No specific setup is required for PeopleSoft GL and Oracle Retail. However, you must perform these setup tasks specific to the integration layer:

1. Modify the required domain value maps (DVMs) with valid data.
The installation process exports DVMs to the location specified during installation.
2. Create these views in the Oracle Retail database: IM_FINANCIALS_STG_VW and FIF_STG_GL_DATA_VW.
3. Set up the AIAConfigurationProperties.XML file.
Set up only the email properties and data polling time.

4.3 Configuring and Generating Data in Oracle Retail

This section discusses these topics:

- [Section 4.3.1, "Configuring the Data in Oracle Retail"](#)
- [Section 4.3.2, "Generating the Data in Oracle Retail"](#)

4.3.1 Configuring the Data in Oracle Retail

Before running any accounting data, perform these tasks in Oracle RMS and Oracle ReIM:

- Set up a GL account cross-reference.
- Define GL options.

For more information, see the *RMS User Guide*, "Financial Management," "Stock Ledger," "Maintain general ledger cross-reference" and the *ReIM User Guide*, "System Administration," "General Ledger Accounts."

4.3.2 Generating the Data in Oracle Retail

Oracle RMS stages GL data for subsequent upload into the integrated financial system. A set of batch processes gather and organize the data before using it to populate the related staging table.

These batch designs are included in this functional area:

- FIFGLDN1.PC
- FIFGLDN1.PC
- FIFGLDN3.PC

For more information, see the *RMS Operations Guide*, "General Ledger (GL) Batch."

In Oracle ReIM, the batch process engages in these high-level steps:

1. Performs any resolution actions (for example, initiates the creation of payment documents).
2. Calls the posting process to write applicable financial accounting transactions to the financials staging table, IM_FINANCIALS_STAGE.

The processing occurs after discrepancies for documents have been resolved by resolution documents. After all of the resolution documents for a matched invoice are built and all of the RCA and RUA external processing has been confirmed, the process inserts financial accounting transactions to the financials staging table to represent the resolution and consequent posting of the invoice. The process also inserts financial accounting transactions for the approved documents that are being handled.

After all of the transactions have been written, the process switches the status of the current invoices or documents to *Posted* and then moves on to the next invoice or document. If a segment lookup fails, the system writes the failed record to a financials error table.

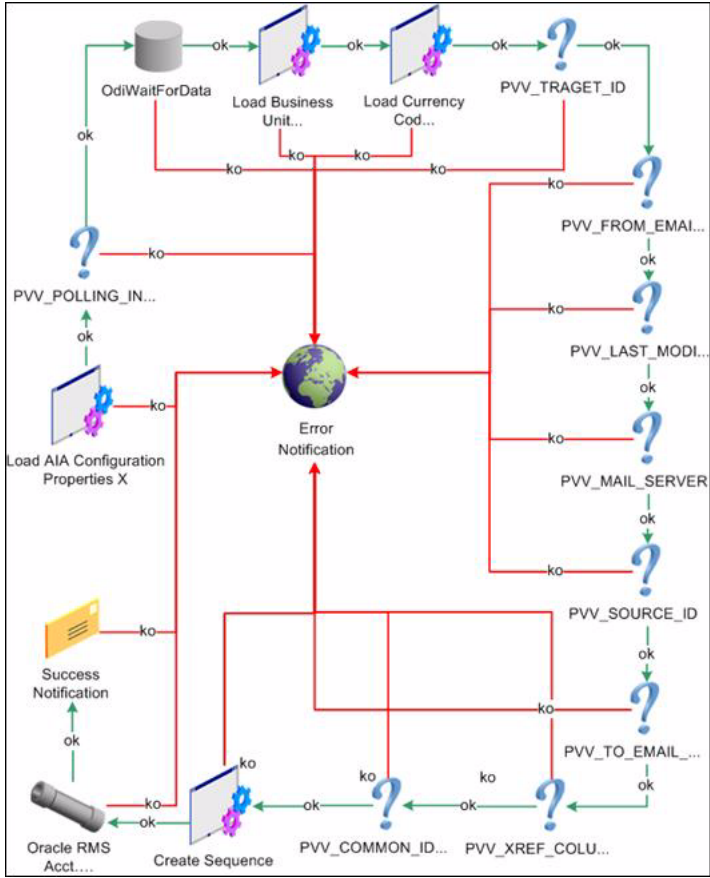
For more information, see the *ReIM Operations Guide*, "Batch Processes," Resolution posting action rollup batch design.

4.4 Picking Up and Transforming the Data

The process integration for accounting entries uses ODI to pick up the data from Oracle RMS, Oracle ReSA, and Oracle ReIM, transform it, and load it into the PeopleSoft staging table for the GL Generator.

[Figure 4-2](#) illustrates the process.

Figure 4-2 Accounting entries load process flowchart



Perform these tasks as part of the accounting entries load process:

1. This process loads the AIAConfigurationProperties.XML file into the ODI system.
2. The system polls the STG_FIF_GL_DATA or IM_FINANCIALS_STG tables (depending on whether the Oracle RMS or Oracle ReIM package is running) until the data has been loaded in the tables.
The rate of polling depends upon a property set in the AIAConfigurationProperties.XML file.
3. ODI loads the BUSINESS_UNIT.XML and CURRENCY_CODE.XML DVM files into the system for mapping purposes.
4. The system loads all user-defined properties and variables from the AIAConfigurationProperties.XML file.
5. The system creates a database sequence to uniquely number entries in the PS_ORT_ACCT_ENTRY table in PeopleSoft, if the database is not created.
6. An ODI interface object is run to transform and map the data from the Retail table to the PSFT table.
7. If the package is successful, an email confirmation is sent to the administrator indicating the number of successfully loaded records.

Note: If any of these steps fails, an error message is sent to the AIAAsyncErrorHandlerBPELProcess, and the data is rolled back into the Retail table, saving it for another run after the error is fixed.

4.5 Configuring and Running the Process Integration for Accounting Entries

This section discusses these topics:

- [Section 4.5.1, "Configuring the Process Integration for Accounting Entries"](#)
- [Section 4.5.2, "Setting Up a Schedule"](#)
- [Section 4.5.3, "Running the Process Integration for Accounting Entries"](#)

4.5.1 Configuring the Process Integration for Accounting Entries

After you have installed the process integration for accounting entries, you must configure several parameters. Open the AIAConfigurationProperties.XML file that is available in the AIA_HOME>/aia_instances/\$INSTANCE_NAME/AIAMetaData/config folder.

For more information about how to update MDS, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Building AIA Integration Flows," Updating MDS.

For more information about requirements for working with AIAConfigurationProperties.xml, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Building AIA Integration Flows," How to Set Up AIA Workstation.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

4.5.1.1 SettingUpAIAConfigurationProperties.XML

When the process integration for accounting entries completes successfully, the system sends an email confirmation containing the number of successfully loaded records. You can configure the email properties in the AIAConfigurationProperties.XML file.

Under the module name RetailToPeopleSoft_AccountingEntry, update these properties:

```
<Property name="FromMailAddress">emailFrom</Property >
<Property name="ToMailAddress">emailTo</Property>
<Property name="MailServerName">mail.oracle.com</Property>
```

For the FromMailAddress property, configure an email address to be used as the sender's email address for email notification.

For the ToMailAddress property, configure an email address where the notification is sent.

For the MailServerName property, configure the valid mail server that is used to send email notifications.

For this process integration for accounting entries, the system polls the staging tables until the tables are populated with the data. You can configure the rate of polling property in the AIAConfigurationProperties.XML file.

If desired, update the PollingInterval property:

```
<Property name="PollingInterval">60000</Property>
<Property name="FromMailAddress">john.doe@oracle.com</Property>
```

For this process integration for accounting entries, the system requires a database sequence to uniquely number entries in the PS_ORT_ACCT_ENTRY table in PeopleSoft.

If desired, update the CommonIDSeq property:

```
<Property name="CommonIDSeq">PS_AIA_COMMON_ID_SEQ</Property>
```

4.5.1.2 Setting Up and Exporting DVMs

During the installation process, DVMs are exported automatically to the location specified in Oracle Universal Installer (OUI). However, you must modify these DVM with the correct data:

- BUSINESS_UNIT
- CURRENCY_CODE

Set up and export the CHARTOFACCOUNTS_GLELEMENT DVM to <AIA_HOME>/services/core/BulkDataProcess/OracleRetailToPeopleSoft/Financials/CreateRetailViewSQL.

4.5.1.3 Creating Views in Retail Database

Whenever a set of books ID (business unit) is created in the Oracle Retail and PeopleSoft systems, you must regenerate the views for including the new set of books ID.

To create the views in the Oracle Retail database, complete these steps:

1. Set up the cross-reference between segments and ChartFields.
 - A. Update the DVM:
 - a. Navigate to SOA_HOME/common/bin:
/slot/ems3344/oracle/Middleware/Oracle_SOA1/common/bin
 - b. Open the WebLogic scripting tool: ./wlst.sh
 - c. Connect to your server using this command:
connect('weblogic','weblogic#1','t3://sdc60012sems.us.oracle.com:7045')
 - d. To export a single DVM, use this command:
exportMetadata(application='soa-infra',server='soa_server1',toLocation='/slot/ems3344/oracle/ODI/ExportedDVMs',docs='/apps/AIAMetaData/dvm/SUPPLIERPARTY_STATUSCODE.dvm')
 - e. To export all DVMs, use this command:
exportMetadata(application='soa-infra',server='soa_server1',toLocation='/slot/ems3344/oracle/ODI/ExportedDVMs',docs='/apps/AIAMetaData/dvm/**')
 - f. To leave WLST, use this command: exit()
 - g. If you put your toLocation as /slot/ems3344/oracle/ODI/ExportedDVMs, it is appended as follows:
/slot/ems3344/oracle/ODI/ExportedDVMs/apps/AIAMetaData/dvm

- h. To export your AIAConfigurationProperties file, use this command:
`exportMetadata(application='soainfra',server='soa_server1',toLocation='/slot/ems3344/oracle/ODI/ExportedDVMs',docs='/apps/AIAMetaData/config/AIAConfigurationProperties.xml')`
 - i. change .dvm file to .xml
- B.** Export the CHARTOFACCOUNTS_GLELEMNT DVM into the <AIA_HOME>/services/core/BulkDataProcess/OracleRetailToPeopleSoft/Financials/CreateRetailViewSQL folder.
2. Run the Java program to generate the SQL scripts.
 - a. From the command line, navigate to the CreateRetailViewSQL folder.
 - b. Call the ANT script CreateRetailView.xml.
 This ANT script generates these SQL scripts: STG_FIF_GL_DATA_VW.sql IM_FINANCIALS_STG_VW.sql IM_AP_STAGE_DETAIL_VW.sql
 - c. Run these SQL scripts into the Oracle Retail database to create the views: STG_FIF_GL_DATA_VW.sql, IM_FINANCIALS_STG_VW.sql, and IM_AP_STAGE_DETAIL_VW.sql.

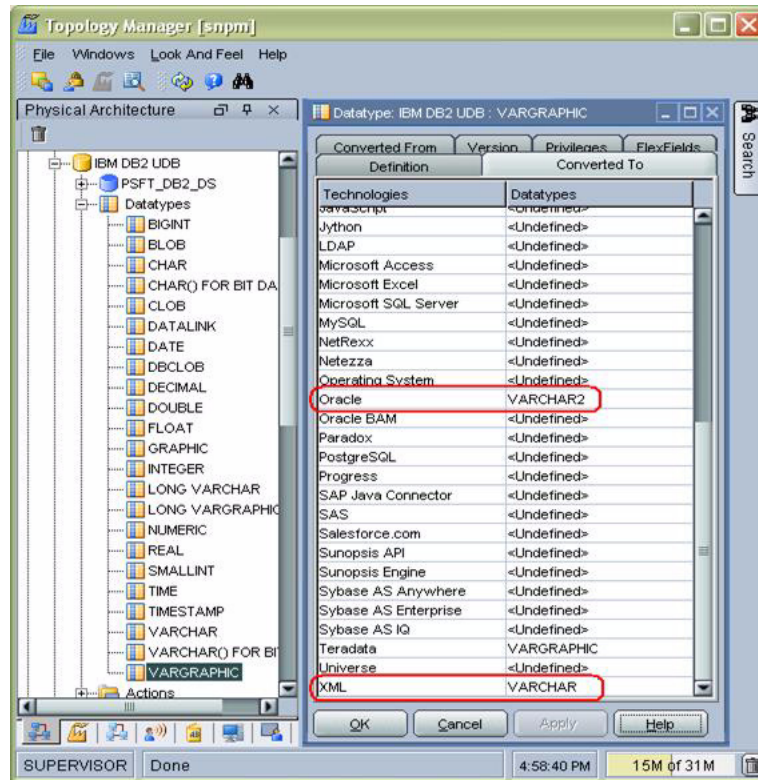
4.5.2 Setting Up a Schedule

A package is the main processing unit in ODI. A package is made of a sequence of steps organized in an execution diagram. This integration includes these packages:

- If the PeopleSoft application is on an Oracle database:
 - Load RMS To PSFT Acct Entry Package
 - Load ReIM To PSFT Acct Entry Package
- If the PeopleSoft application is on an IBM DB2 database:
 - Load RMS To PSFT DB2 Package
 - Load ReIM To PSFT DB2 Package

If the PeopleSoft system is running on the IBM DB2 database, set up the Datatype conversion VARGRAPHIC for the Oracle and XML technology, as illustrated in [Figure 4-3](#), before generating the scenario.

Figure 4-3 Example of the Datatype conversion to VARGRAPHIC setup



For each package, users must generate a scenario and then schedule it. A scenario is the execution unit for production that can be scheduled.

For more information about creating schedules and scenarios, see *Oracle Data Integrator User Guide* and *Oracle Data Integrator Installation Guide*. These documents are available in <odi install location?>\oracledi\doc subdirectory.

4.5.3 Running the Process Integration for Accounting Entries

This process runs based on the schedules created for it. You can schedule the integration between Oracle Retail and PeopleSoft GL to occur automatically.

Note: You can start or stop the scheduler agent whenever you want, which enables you to control the integration process.

For more information about starting and stopping the scheduler agent, see *Oracle Data Integrator User Guide*, "Launching a Scheduler Agent" and "Stopping an Agent." This document is available in the <odi install location?>\oracledi\doc subdirectory.

Process Integration for Retail Merchandise Procure to Pay

This chapter provides an overview of the process integration for Retail Merchandise Procure to Pay between Oracle Retail Invoice Matching (Oracle ReIM) and PeopleSoft Payables and discusses how to perform setup tasks, configure and generate data in Oracle Retail, pick up and transform the data, and configure and run the process integration for retail merchandise procure to pay.

This chapter includes the following sections:

- [Section 5.1, "Process Integration for Retail Merchandise Procure to Pay Overview"](#)
- [Section 5.2, "Performing Setup Tasks"](#)
- [Section 5.3, "Configuring and Generating Data in Oracle Retail"](#)
- [Section 5.4, "Picking Up and Transforming the Data"](#)
- [Section 5.5, "Configuring and Running the Process Integration for Retail Merchandise Procure to Pay"](#)

5.1 Process Integration for Retail Merchandise Procure to Pay Overview

Oracle ReIM is the source of matched invoices, credit notes, debit memos, and rebates. PeopleSoft Payables requires these details for payment to suppliers. Invoices from suppliers for retail merchandise are matched to the original purchase order (PO) for the merchandise and the receipt of the merchandise by the retailer. A proper match of invoice, PO, and receipt trigger the payment authorization of the supplier's invoice. Invoices may be authorized for payment before receipt of goods for which prepayment is required. When the authorization for payment is generated, the appropriate accounting distribution is also generated to support the payment authorization. The Retail Merchandise Procure to Pay integration automates the processing of invoice payments, adjustments, and write-offs from Oracle ReIM to PeopleSoft Payables and General Ledger. Other accounting transactions are generated from Oracle ReIM to write off aged receipts that were never invoiced and to post accounting distributions for manually paid or prepaid invoices after goods or services are received.

This integration uses the Oracle Data Integrator (ODI) application to transfer the data from Oracle ReIM to PeopleSoft Payables.

The process integration between Oracle ReIM and PeopleSoft Payables supports these integration flows:

- Post matched invoices for payment from Oracle ReIM to PeopleSoft Payables.

- Post credit notes (matched or unmatched) for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post debit or credit memos for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post rebates for payment adjustment from Oracle ReIM to PeopleSoft Payables.
- Post unmatched invoices for prepayment from Oracle ReIM to PeopleSoft Payables.

5.1.1 Business Process Flow for Integration of Retail Merchandise Procure to Pay

Figure 5–1 illustrates the process flow for Retail Merchandise Procure to Pay integration.

Figure 5–1 Business process flow for Retail Merchandise Procure to Pay

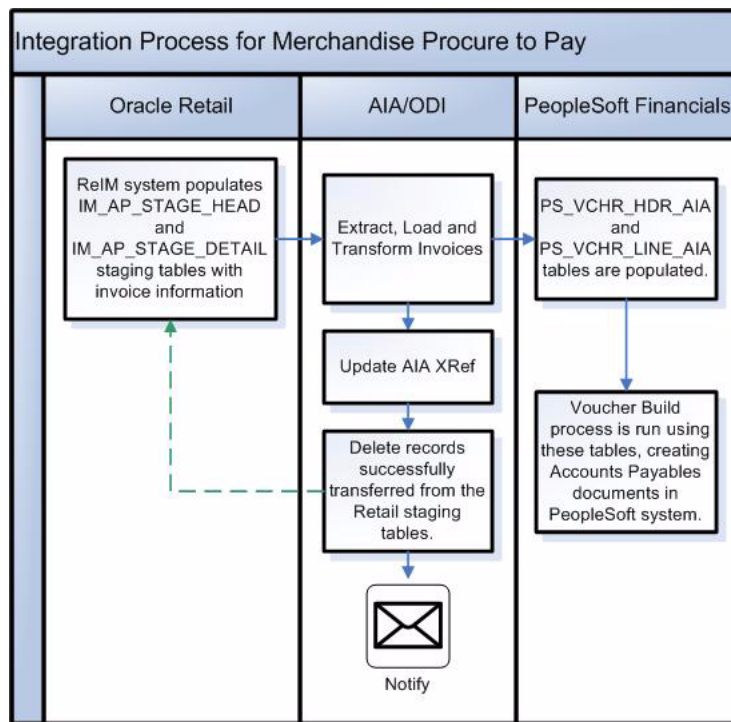


Table 5–1 describes the overall process.

Table 5–1 Retail Merchandise Procure to Pay Overall Process

Work Location	Step
Oracle Retail (ReIM)	1. Configure and generate outgoing data from Oracle ReIM.

Table 5–1 (Cont.) Retail Merchandise Procure to Pay Overall Process

Work Location	Step
Integration process	<p>2. The integration process checks for data in the Oracle ReIM Invoice Header and Invoice Lines interface tables.</p> <p>3. Populate the Invoice cross-reference with the Retail key and common key.</p> <p>4. Load and transform the Oracle Retail data into the PeopleSoft Voucher Header and Voucher Lines Staging tables.</p> <p>5. Delete the data from the source tables (Oracle ReIM Invoice Header and Invoice Lines interface tables) after a successful load.</p>
PeopleSoft Payables	<p>6. Run the Voucher Build process and import the vouchers into the PeopleSoft Payables system for the pay cycle.</p> <p>7. Invoke a process to populate the new PeopleSoft Invoice cross-reference table with the PSFT key and RETL key.</p>

5.1.2 Solution Assumptions and Constraints

The integration design assumes that:

1. Before the system imports the invoices from Oracle Retail to PeopleSoft Payables, these integrations are completed successfully:
 - a. Supplier synchronization and supplier cross-reference.
 - b. Supplier address synchronization and supplier address cross-reference.
 - c. Payment terms synchronization and payment terms cross-reference.
 - d. Currency code domain value map (DVM) and currency exchange rate synchronization.
 - e. SetID, business unit, and ChartFields or segment DVM.
2. Chart of accounts combinations are manually entered into the Oracle Retail and PeopleSoft applications.
3. The DVM data is set up in the Service Oriented Architecture (SOA) suite and exported from the SOA suite that is used as input in the ODI interfaces.
4. The data moves from one instance of source database to one instance of target database.
5. If the user loads the same data twice in the Oracle Retail interface table, then the data is transported again to the PeopleSoft interface tables.

In the ODI interface, no business validation is applied to check whether any data is transported.
6. The chart of accounts combinations are valid for invoice lines.

Invoice lines with invalid combinations are rejected by the PeopleSoft Voucher Build process, and the correction of the same is a manual process in PeopleSoft applications.
7. PeopleSoft GL does not delete the invoices with source as RETL.

8. Oracle Retail modifies the IM_AP_STAGE_HEAD and IM_AP_STAGE_DETAIL tables and adds segment 11 through segment 20 for the dynamic mapping to PeopleSoft ChartFields.
9. The monetary amount fields in the PeopleSoft and Oracle Retail systems have a mismatch in the number of digits after the decimal.

The PeopleSoft system allows only 3 digits after a decimal, and Oracle Retail allows 4 digits. Therefore, some data may be rounded while moving from Oracle Retail to PeopleSoft.

5.2 Performing Setup Tasks

This section discusses the following topics:

- [Section 5.2.1, "Setup Tasks Specific to Oracle ReIM"](#)
- [Section 5.2.2, "Setup Tasks Specific to PeopleSoft Payables"](#)
- [Section 5.2.3, "Setup Tasks Specific to the Integration Layer"](#)

5.2.1 Setup Tasks Specific to Oracle ReIM

For more information about the setup tasks specific to Oracle ReIM, see the *ReIM Operations Guide*, "PeopleSoft Enterprise Financials Integration."

5.2.2 Setup Tasks Specific to PeopleSoft Payables

Perform these tasks:

- Activate appropriate service operations, handlers, routings, and queues and define the appropriate security groups for the service operations.
- Populate DVMs with appropriate PeopleSoft and common values.
- Define the end points for outbound services.
- Define recurring run controls for repetitively scheduled processes.

For more information about these tasks, see [Section 7.4, "Setting Up PeopleSoft GL."](#)

5.2.3 Setup Tasks Specific to the Integration Layer

Perform these tasks:

- Synchronize suppliers and their locations.
- Synchronize payment terms.
- Map the chart of accounts segment between the Oracle Retail and PeopleSoft applications.
- Validate the chart of accounts code combination.
- Synchronize the currency exchange rates.
- Modify these DVMs:
 - CURRENCY_CODE DVM
 - BUSINESS_UNIT DVM

These DVMs are exported during installation to the directory specified in Oracle Universal Installer (OUI).

- Generate IM_AP_STAGE_DETAIL_VW:
 - Modify the COA_SEGMENT_MAP DVM in Oracle JDeveloper.
 - Export the DVM and save the XML file.
 - Verify the IM_AP_STAGE_DETAIL_VW.ini file for the view creation.
 - Run the CreateViewText.java program and pass the parameters as IM_AP_STAGE_DETAIL_VW.ini COASegmentMapDVM.xml IM_AP_STAGE_DETAIL_VW.sql 'RETL_01' 'PSFT_01'.
 - Verify the IM_AP_STAGE_DETAIL_VW.sql file and create the view in the Oracle Retail database.
- Modify the AIAConfigurationProperties.XML file with the parameters shown in [Example 5-1](#).

Example 5-1 AIAConfigurationProperties.XML Parameters

```
<ModuleConfiguration moduleName="RetailToPeopleSoftInvoiceIntegration">
  <Property name="ToMailAddress">john.doe@oracle.com</Property>
  <Property name="MailServerName">mail.oracle.com</Property>

  <!-- **-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
    The process waits for the tables to be populated, polling at x milliseconds
    intervals.
    **-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*
  -->
  <!-- The period in milliseconds to wait between data polls. -->
  <Property name="PollingInterval">60000</Property>

  <!-- Path to exported DVMS -->
  <Property name="DVMPATH">C:\YourDVMLocation</Property>

  <!-- Name of the DVM Column and XREF_DATA.ColumnName for Retail values. -->
  <Property name="SenderID">RETL_01</Property>

  <!-- Name of the DVM Column and XREF_DATA.ColumnName for PeopleSoft values. -->
  <Property name="TargetID">PSFT_01</Property>

  <!-- Name to use for Table Name in the XREF_DATA. -->
  <Property name="XRefTableName">PAYABLEINVOICE_PAYABLEINVOICEID</Property>
</ModuleConfiguration>
```

5.3 Configuring and Generating Data in Oracle Retail

This section discusses the following topics:

- [Section 5.3.1, "Configuring the Data in Oracle Retail"](#)
- [Section 5.3.2, "Generating the Data in Oracle Retail"](#)

5.3.1 Configuring the Data in Oracle Retail

Before running any transaction data, complete these tasks in Oracle ReIM:

- Set up General ledger (GL) account cross-reference.
- Define GL options.

For more information, see the *ReIM User Guide*, "System Administration," General Ledger Accounts.

5.3.2 Generating the Data in Oracle Retail

In Oracle ReIM, the batch process engages in these high-level steps:

1. Performs any resolution actions (for example, instigates the creation of payment documents).
2. Calls the posting process to write applicable financial accounting transactions to the financials staging table, `IM_FINANCIALS_STAGE`.

The processing occurs after resolution documents resolve discrepancies for documents. After all of the resolution documents for a matched invoice are built and all of the RCA and RUA external processing is confirmed, the process inserts financial accounting transactions into the financials staging table to represent the resolution and consequent posting of the invoice. The process also inserts financial accounting transactions for the approved documents that are being handled.

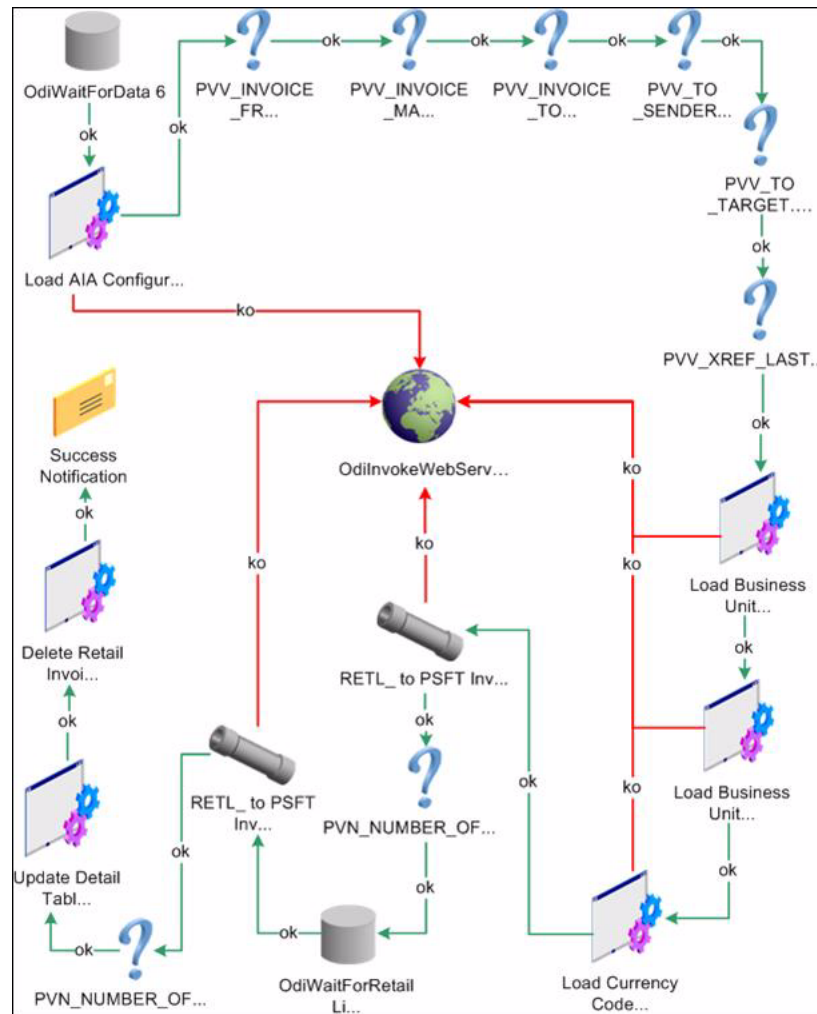
After all of the transactions are written, the process switches the status of the current invoices or documents to *Posted* and then moves on to the next invoice or document. If a segment lookup fails, the system writes the failed record to a financials error table.

For more information, see the *ReIM Operations Guide*, "Batch Processes," Resolution posting action rollup batch design.

5.4 Picking Up and Transforming the Data

The process integration for Retail Merchandise Procure to Pay uses ODI to pick up the data from Oracle ReIM, transform the data, and load the transformed data into the PSFT staging tables for the Voucher Build process.

[Figure 5-2](#) illustrates the process.

Figure 5–2 Retail Merchandise Procure to Pay process flow

Complete these tasks as part of the Retail Merchandise Procure to Pay integration:

1. The Oracle Retail system polls the IM_AP_STAGE_HEAD table until the table is populated with data.
2. The system loads the AIAConfigurationProperties.XML file into the ODI system.
3. The system loads all user-defined properties and variables from the AIAConfigurationProperties.XML file into Oracle ReIM.
4. ODI loads the BUSINESS_UNIT.xml and CURRENCY_CODE.XML DVM files into Oracle ReIM for mapping.
5. An ODI interface object is run to transform and map the data from the Retail IM_AP_STAGE_HEAD table to the PeopleSoft PS_VCHR_HDR_AIA table.
6. A variable is set to capture the number of records that were inserted for the header interface.
This variable is used in the email confirmation.
7. An ODI interface object is run to transform and map the data from the Retail IM_AP_STAGE_DETAIL table to the PeopleSoft PS_VCHR_LINE_AIA table.

8. A variable is set to capture the number of records that were inserted for the detail interface.

This variable is also used in the email confirmation.

9. Two procedures are run. The first updates the Retail IM_AP_STAGE_DETAIL table with the ODI session number.

The second deletes the records from both the Retail header and detail tables based on this ODI session number. The ODI session is automatically populated in the header table by the header interface.

Note: If any of these steps fails, an error message is sent to the AIAAsyncErrorHandler BPEL process, and the data is rolled back in the Retail tables, saving it for another run after the error is fixed.

5.5 Configuring and Running the Process Integration for Retail Merchandise Procure to Pay

This section discusses the following topics:

- [Section 5.5.1, "Configuring the Process Integration for Retail Merchandise Procure to Pay"](#)
- [Section 5.5.2, "Setting Up a Schedule"](#)
- [Section 5.5.3, "Running the Process Integration for Retail Merchandise Procure to Pay"](#)

5.5.1 Configuring the Process Integration for Retail Merchandise Procure to Pay

After you have installed the process integration for Retail Merchandise Procure to Pay, you must configure several parameters. Open the AIAConfigurationProperties.XML file that is available in the AIA_HOME>/aia_instances/\$INSTANCE_NAME/AIAMetaData/config folder.

For more information about requirements for working with AIAConfigurationProperties.xml, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Building AIA Integration Flows," How to Set Up AIA Workstation.

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

5.5.1.1 Setting Up AIAConfigurationProperties.XML

When the process integration for Retail Merchandise Procure to Pay completes successfully, the system sends an email confirmation containing the number of successfully loaded records. You can configure the email properties in the AIAConfigurationProperties.XML file.

Under the moduleName RetailToPeopleSoftInvoiceIntegration, update these properties:

```
<Property name="FromMailAddress">emailFrom</Property>
<Property name="ToMailAddress">emailTo</Property>
<Property name="MailServerName">mail.oracle.com</Property>
```

For the FromMailAddress property, configure an email address to be used as the sender's email address for email notification.

For the ToMailAddress property, configure an email address where the notification is sent.

For the MailServerName property, configure the valid mail server that is used to send email notifications.

For this process integration, the system polls the staging tables until the tables are populated with the data. You can configure the rate of polling property in the AIAConfigurationProperties.XML file.

If desired, update the PollingInterval properties:

```
<Property name="PollingInterval">60000</Property>
<Property name="PSFTUrl.ProtocolHostPort"></Property>
```

For this process integration, the system requires a database sequence to uniquely number entries in the PS_ORT_ACCT_ENTRY table in PeopleSoft applications.

If desired, update the CommonIDSeq property:

```
<Property name="CommonIDSeq">PS_AIA_COMMON_ID_SEQ</Property>
```

5.5.1.2 Setting Up and Exporting DVMs

During the installation process, DVMs are exported automatically to the location specified in OUI. However, you must modify these DVMs with the correct data:

- BUSINESS_UNIT
- CHARTOFACCOUNTS_GLELEMENT
- CURRENCY_CODE

5.5.1.3 Creating Views in Retail Database

Whenever a set of books ID (business unit) is created in the Oracle Retail and PeopleSoft systems, you must regenerate the views to include the new set of book IDs.

Complete these steps to create views in the Oracle Retail database:

1. Set up the cross-reference between segments and ChartFields.
 - A. Update the DVM:
 - a. Log in to the Oracle Fusion Middleware (FMW) server using SSH.
 - b. Open the WSLT console. The commands to start the console are as follows.


```
cd <SOA_HOME>/common/bin
./wlst.sh
```
 - c. Connect to the SOA server using the connect command: connect('<SOA_USER_NAME>', '<SOA_USER_PASSWORD>', 't3://<SOA_HOST>:<SOA_ADMIN_PORT>')

Example:

```
connect('weblogic','weblogic#1','t3://sdc60011sems.us.oracle.com:7044')
```
 - d. Export the required DVM using the exportMetadata command:


```
exportMetadata(application='<APPLICATION_NAME>',server='<SOA_SERVER>',toLocation='<TARGET_PATH>',docs='<DVM_PATH>')
```

Example: `exportMetadata(application='soa-infra',server='soa_server1',toLocation='/slot/ems5343/oracle/export',docs='/apps/AIAMetadata/dvm/ITEM_INDICATOR.dvm')`

- e. Exit the WSLT console using the `exit()` command.
- B. Export CHARTOFACCOUNTS_GLELEMNT DVM into the `<AIA_HOME>/apps/services/core/BulkDataProcess/OracleRetailToPeopleSoft/Financials/CreateRetailViewSQL`.
2. Run the Java program to generate the SQL scripts.
 - a. From the command line, navigate to the `CreateRetailViewSQL` folder.
 - b. Call the ANT script `CreateRetailView.xml`.
 This ANT script generates these SQL scripts: `STG_FIF_GL_DATA_VW.sql`, `IM_FINANCIALS_STG_VW.sql`, and `IM_AP_STAGE_DETAIL_VW.sql`.
3. Run these SQL scripts in the Oracle Retail database to create the views: `STG_FIF_GL_DATA_VW.sql`, `IM_FINANCIALS_STG_VW.sql`, and `IM_AP_STAGE_DETAIL_VW.sql`.

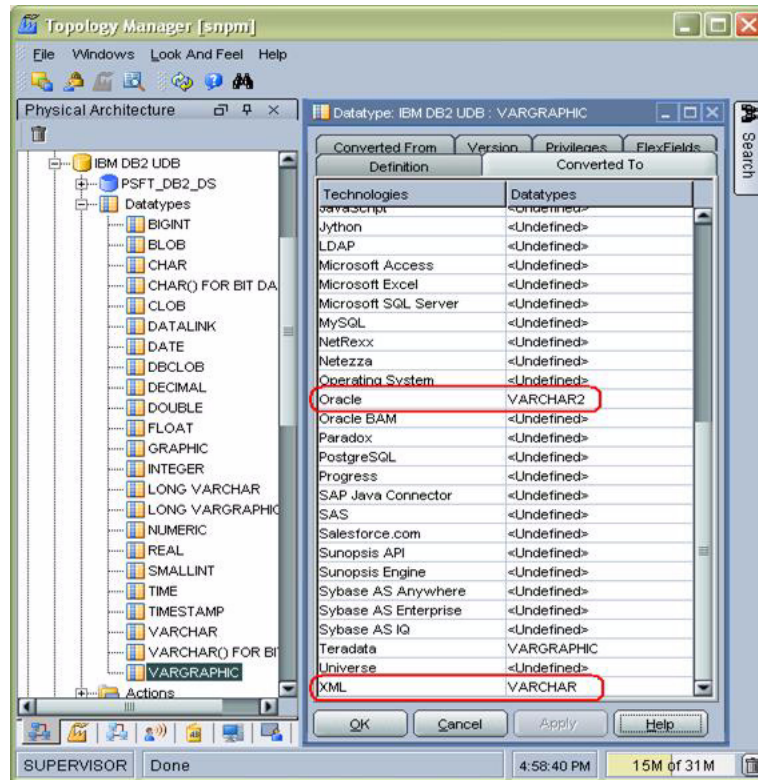
5.5.2 Setting Up a Schedule

A package is the main processing unit in ODI. A package is made of a sequence of steps organized in an execution diagram. This integration includes these packages:

- If PeopleSoft applications are on an Oracle database: Load RETL Invoices To PSFT Package
- If PeopleSoft applications are on an IBM DB2 database: Load RETL Invoices To PSFT DB2

If the PeopleSoft system is running on the IBM DB2 database, set up the Datatype conversion to VARGRAPHIC for the Oracle and XML technology, as illustrated in [Figure 5-3](#), before generating the scenario:

Figure 5–3 Example of the setup of the Datatype conversion to VARGRAPHIC



For each package, users must generate a scenario and then schedule it. A scenario is the execution unit for production that you can schedule.

For more information about creating schedules and scenarios, see the *Oracle Data Integrator User Guide* and the *Oracle Data Integrator Installation Guide*. These documents are available in <odi install location?>\oracledi\doc subdirectory.

5.5.3 Running the Process Integration for Retail Merchandise Procure to Pay

This process runs based on schedules you create for it. You can schedule the integration between Oracle Retail and PeopleSoft Payables to occur automatically.

Note: You can start or stop the scheduler agent whenever you want, which enables you to control the integration process.

For more information about starting and stopping the scheduler agent, see *Oracle Data Integrator User Guide*, "Launching a Scheduler Agent" and "Stopping an Agent." This document is available in <odi install location?>\oracledi\doc subdirectory.

Drill Back and Drill Forward

This chapter provides an overview of the drill back and drill forward actions between the Oracle Retail and PeopleSoft applications and discusses drill back and drill forward details, and service solution components.

This chapter includes the following sections:

- [Section 6.1, "Drill Back and Drill Forward Overview"](#)
- [Section 6.2, "Drill Back and Drill Forward Details"](#)
- [Section 6.3, "Service Solution Components"](#)

6.1 Drill Back and Drill Forward Overview

Drill back and drill forward enables users of PeopleSoft Financials and Oracle Retail applications to access information about the financial transactions from the integrated systems in a seamless and intuitive manner. It provides the users the end-to-end traceability between PeopleSoft Payables and General Ledger back to the Oracle Retail Merchandising System (Oracle RMS), Oracle Retail Sales Audit (Oracle ReSA), and Oracle Retail Invoice Matching (Oracle ReIM). Drill back and drill forward supports navigation at the user interface level from the PeopleSoft applications to the Oracle Retail applications and vice versa for the transactions that are integrated between the two systems.

The drill back and drill forward feature enables PeopleSoft Financials and Oracle Retail application users to trace and investigate financial transactions by:

- Drilling back from a journal entry in PeopleSoft General Ledger (GL) to Oracle RMS, Oracle ReSA, and Oracle ReIM to find the source of the journal or from a voucher in PeopleSoft Payables to an invoice in Oracle ReIM.
- Drilling forward from Oracle RMS, Oracle ReSA, or Oracle ReIM to PeopleSoft GL and Payables for detailed information about posted transactions for a source transaction.

In this integration, the Oracle Application Integration Architecture (Oracle AIA) layer maintains the cross-references to associate the object identifications among the Oracle RMS, Oracle ReSA, and Oracle ReIM systems and the PeopleSoft GL and Payables systems.

The cross-reference is built during the processes for submission of invoices, credit notes, debit memos, and rebates in Oracle ReIM, Oracle RMS, and Oracle ReSA to PeopleSoft Payables. The Oracle AIA services use the cross-references to link the invoices to payments.

These Oracle Retail and PeopleSoft applications participate in the integrated environment:

- Oracle RMS Stock Ledger: This application is the source of transactions for the valuation of sellable inventory located in stores, warehouses, and distribution centers.
- Oracle ReSA: This application is the source of transactions for recognition of revenue and cash posting.
- Oracle ReIM: This application is the source of invoices to be paid, credit adjustments, debit memos, write-offs, and invoice adjustments.
- PeopleSoft GL: This application posts the accounting entries from the Oracle Retail applications (Oracle RMS, Oracle ReSA, and Oracle ReIM) to designated ledgers.
- PeopleSoft Payables: This application processes invoices for payment and posts the results in PeopleSoft GL based on the templates and account coding.

The drill back and drill forward actions support these flows:

- **Drill back from ledger entry to revenue recognition transactions** for information about revenue recognition transactions using the ledger entry.
- **Drill back from ledger entry to inventory valuation transactions** for information about inventory valuation transactions using the ledger entry.
- **Drill back from ledger entry to aged open receipt write-off transactions** for information about aged open-receipt write-off transactions using the ledger entry.
- **Drill back from ledger entry to matched prepaid invoice transactions** for information about matched pre-paid invoice transactions using the ledger entry.
- **Drill back from payment voucher to matched invoice, credit notes, debit memos, and rebates** for details of matched invoice, credit notes, debit memos, and rebates from the related payment voucher.
- **Drill forward from sales audit transaction to ledger** for details of a particular ledger entry using the sales audit transaction information.
- **Drill forward from stock ledger transaction to ledger** for details of a particular ledger entry using the stock ledger transaction information.
- **Drill forward from matched invoice to payment voucher** for details of a particular payment voucher from the related matched invoice.
- **Drill forward from credit note, debit memo, or rebate to payment voucher** for details of a particular payment voucher from the related credit note, debit memo, or rebate.
- **Drill forward from aged open receipt write-off transactions to ledger** for details of a particular ledger entry from the aged open receipt write-off transaction information.
- **Drill forward from matched prepaid invoice to ledger** for details of a particular ledger entry from the matched prepaid invoice information.

6.1.1 Setup Activities

Modify the AIAConfigurationProperties.XML file with the parameters shown in [Example 6-1](#).

Example 6–1 AIAConfigurationProperties Parameters

```
<ServiceConfiguration serviceName="DrillBackForward">
  <Property name="PSFTUrl.Servlet"></Property>
  <Property name="PSFTUrl.Site"></Property>
  <Property name="PSFTUrl.PortalNodeContent"></Property>
  <Property name="PSFT.Invoice.WS.Location"></Property>
  <Property name="RETAIL.URL.WS.Location"></Property>
</ServiceConfiguration>
```

All the values for these properties are populated automatically with information obtained from users through Oracle Universal Installer (OUI) screens. Users can also change them directly in this property file.

The first four properties represent the PeopleSoft application URL. The last two represent the PeopleSoft web service location and the Oracle Retail web service location, respectively.

For more information, see *Oracle Fusion Middleware Installation and Upgrade Guide for Oracle Application Integration Architecture Foundation Pack*.

6.1.2 Solution Assumptions and Constraints

The integration design assumes that:

1. The service configuration section in the AIAConfigurationProperties file for "DrillBackForward" is updated with values.
2. Two systems are supported as target systems for forming the URL: Oracle Retail and PeopleSoft. PeopleSoft configuration is done in the AIAConfiguration file.
3. The cross-reference table is populated with data necessary to perform lookup.

6.2 Drill Back and Drill Forward Details

This section discusses the following topics:

- [Section 6.2.1, "Drill Back Details"](#)
- [Section 6.2.2, "Drill Forward Details"](#)
- [Section 6.2.3, "Overall Flow"](#)

6.2.1 Drill Back Details

The drill back feature enables users to:

- Query Oracle ReIM using the payment voucher ID from the PeopleSoft Payables user interface (UI) to find invoices, credit notes, and debit memos that are used to generate the payment voucher.
- Query Oracle RMS and Oracle ReSA from the PeopleSoft GL UI to find the source transactions that generated the journal entry in the ledger for revenue recognition and cash deposit postings.
- Query Oracle RMS stock ledger from the PeopleSoft GL UI to find the source transactions for inventory valuation postings.
- Query Oracle ReIM from the PeopleSoft GL UI to find the source transactions for aged receipt write-offs, manually paid invoices, and matched prepaid invoices.

This integration uses the DrillBackForwardURLService.

6.2.2 Drill Forward Details

The drill forward feature enables users to:

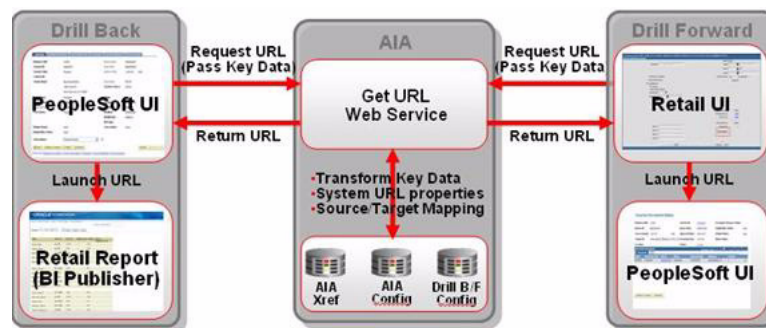
- Query PeopleSoft Payables from the Oracle ReIM UI using the invoice, credit note, debit memo, or document ID to query the resulting payment voucher.
- Query the PeopleSoft GL from Oracle ReIM UI for aged receipt write-offs and prepaid invoices to audit the effect on the ledgers. For aged receipt write-offs, use the receipt ID to access the PeopleSoft GL transaction. For prepaid invoices, use the document ID.
- Query PeopleSoft GL from the Oracle ReSA UI using the sales transaction document ID to audit the journal entry effect of the financials.
- Query PeopleSoft GL from the Oracle RMS UI using the stock ledger document ID or fixed deal reference ID to audit the journal entry effect of the financials.

This integration uses the DrillBackForwardURLService.

6.2.3 Overall Flow

Figure 6–1 illustrates the overall drill back and drill forward flows.

Figure 6–1 Overall flow diagram for drill back and drill forward



When the drill back or drill forward process is initiated:

1. The source application calls the DrillBackForwardURLService web service to create a URL for the target system.
2. The source application passes an array of parameter name and value pairs, and the name of the web service requester (RETAIL or PSFT).
3. The web service determines the target application URL based on the input parameter name and the web service requester name.
4. The cross-reference table and PeopleSoft web service are consulted (as applicable to each case) for parameter values.
5. The DrillBackForwardConfig.xml file is looked up to construct the URL.
6. If the target system is PSFT, the client implementation-specific information (host, port) from the AIAConfiguration.xml file is fetched.
7. A URL with retrieved information is constructed.
8. If the target system is RETAIL, then Oracle Retail web service is called to obtain the URL.
9. The URL is returned to the caller of the service.

10. If an error occurs, a fault message is generated.

6.3 Service Solution Components

Table 6–1 describes the service solution components that are used in the process integration.

Table 6–1 Service Solution Components

Name	Location	Caller	Object
DrillBackForwardURLService	AIA	Oracle Retail or PeopleSoft applications	PAYABLEINVOICE_ PAYABLEINVOICEID or ACCOUNTING_ ENTRY
Invoice Document Report	Retail BI Publisher	PeopleSoft Payables UI	BI Publisher URL
Aged Receipts Transaction Report	Retail BI Publisher	PeopleSoft GL UI	BI Publisher URL
Matched Pre Paid Transaction Report	Retail BI Publisher	PeopleSoft GL UI	BI Publisher URL
Sales Audit Transaction Report	Retail BI Publisher	PeopleSoft GL UI	BI Publisher URL
Stock Ledger Transaction Report	Retail BI Publisher	PeopleSoft GL UI	BI Publisher URL
Stock Ledger Line	Retail BI Publisher	PeopleSoft GL UI	BI Publisher URL

6.3.1 Service Solution Component Details

- DrillBackForwardURLService:** This web service enables users to navigate from the PeopleSoft applications to Oracle Retail (drill back) and Oracle Retail to PeopleSoft applications (drill forward). The source application (Oracle Retail or PeopleSoft application) sends a synchronous request with source system name and an array of parameter name value pairs to this service. This service returns a URL based on the given scenario (drill back or drill forward). The source application launches the returned URL to open the target application in a web browser. If an error occurs, a fault message is generated.
- Invoice Document Report:** PeopleSoft Payables invokes the DrillBackForwardURLService for requesting a report on the invoices, credit notes, debit memos, and rebates from Oracle ReIM that made up the payment voucher. The Oracle AIA layer passes the URL to PeopleSoft Payables UI. PeopleSoft Payables UI initiates a report on the Oracle BI Publisher (Oracle BIP) server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user.
- Aged Receipts Transaction Report:** PeopleSoft GL invokes the DrillBackForwardURLService for requesting a report on the aged receipt write-off transactions generated from Oracle ReIM that made up the journal entry. The Oracle AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the Oracle BIP server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user.
- Matched Pre-Paid Transaction Report:** PeopleSoft GL invokes the DrillBackForwardURLService for requesting a report on the matched pre-paid transactions generated from Oracle ReIM that made up the journal entry. The

Oracle AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the Oracle BIP server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user

- **Sales Audit Transaction Report:** PeopleSoft GL invokes the DrillBackForwardURLService for requesting a report on transactions generated from Oracle ReSA that made up the journal entry. The Oracle AIA layer passes the URL to PeopleSoft GL. PeopleSoft GL UI initiates a report on the Oracle BIP server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user.
- **Stock Ledger Transaction Report:** PeopleSoft GL invokes the DrillBackForwardURLService for requesting a report on inventory valuation transaction generated from Oracle RMS stock ledger that made up the journal entry. The Oracle AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the Oracle BIP server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user.
- **Fixed Deal Transaction Report:** PeopleSoft GL invokes the DrillBackForwardURLService for requesting a report on fixed deal transaction generated from Oracle RMS stock ledger that made up the journal entry. The Oracle AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the Oracle BIP server using the URL created by Oracle AIA. Oracle BIP generates a report for the requesting user.

Figure 6–2 illustrates the solution components for drill back.

Figure 6–2 Solution components for drill back

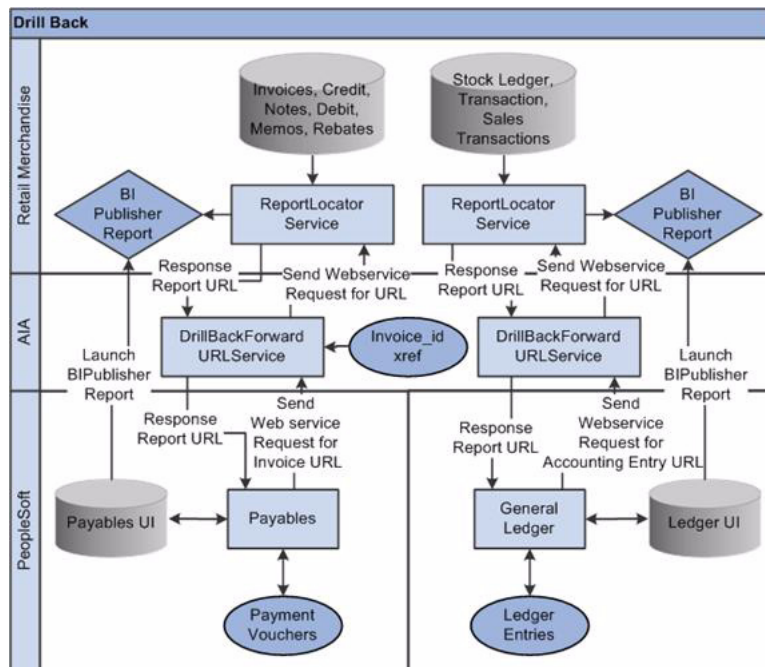
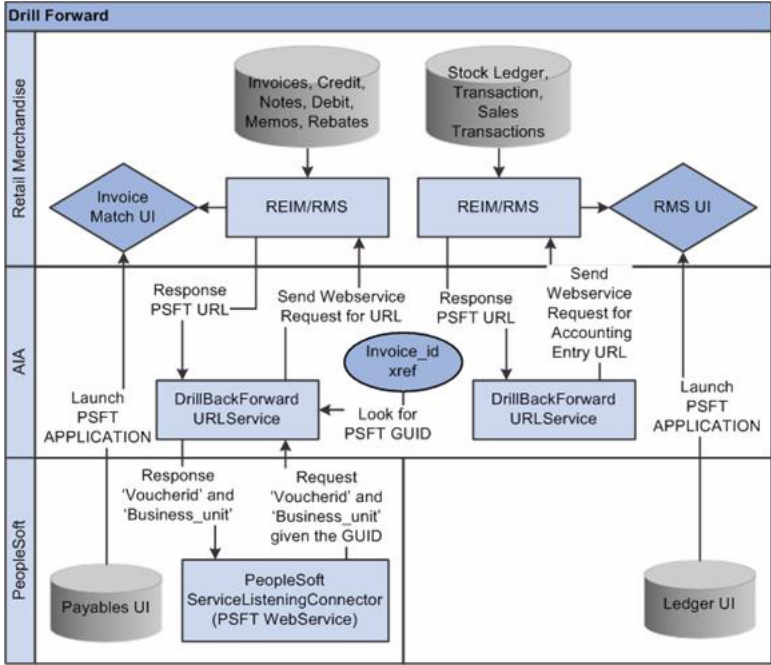


Figure 6–3 illustrates the solution components for drill forward.

Figure 6-3 Solution components for drill forward



Part II

Implementing the Delivered Integrations

Part 2 includes this chapter:

[Chapter 7, "Configuring the Oracle Financial Operations Control Integration Pack"](#)

Configuring the Oracle Financial Operations Control Integration Pack

This chapter discusses how to configure the Oracle Financial Operations Control Integration Pack for Oracle Retail Merchandise Operations Management and PeopleSoft Enterprise Financials.

This chapter includes the following sections:

- Section 7.1, "Prerequisites"
- Section 7.2, "Data Requirements"
- Section 7.3, "Setting Up Oracle Retail Applications"
- Section 7.4, "Setting Up PeopleSoft GL"
- Section 7.5, "Setting Up PeopleSoft Payables"
- Section 7.6, "Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities"
- Section 7.7, "Setting Up PeopleSoft Domain Value Maps"
- Section 7.8, "Manually Synchronizing Freight Terms"
- Section 7.9, "Manually Synchronizing Currency Exchange Rate Types"
- Section 7.10, "Manually Synchronizing Currency Codes"
- Section 7.11, "Manually Synchronizing Chart of Accounts Values"
- Section 7.12, "Manually Synchronizing Financial Calendars"
- Section 7.13, "Mapping Business and Organization Units"
- Section 7.14, "Working with Cross-References"
- Section 7.15, "Working with DVMs"
- Section 7.16, "Handling Errors"
- Section 7.17, "Viewing EIMs"
- Section 7.18, "Setting Configuration Properties"

You must set up Oracle Retail applications, PeopleSoft General Ledger (GL), and PeopleSoft Payables for the Oracle Retail to PeopleSoft process integration pack (PIP) to work properly. This chapter describes these setups in detail.

7.1 Prerequisites

This section discusses the prerequisites for the following integrations:

- Currency exchange rate
- Payment term
- Supplier information
- Inventory valuation and revenue recognition accounting entries
- Retail Merchandise procure to pay
- Drill back and drill forward

7.1.1 Currency Exchange Rate Prerequisites

The prerequisites include:

1. Maintain the list of valid currencies separately in each application and synchronize the currencies manually.
2. Map currency exchange rate types between PeopleSoft applications and Oracle Retail.

Oracle Retail and PeopleSoft applications maintain the exchange rate types manually. Oracle Retail maps the PeopleSoft exchange rate types to the Oracle Retail exchange rates types with a local process.

7.1.2 Payment Term Prerequisites

The process integration for payment term has no prerequisites.

7.1.3 Supplier Information Prerequisites

The prerequisites include:

1. Initially load the ORGANIZATION_ID cross-reference table using the manual process. This step takes place post installation. The values for the xml file come from both the PeopleSoft system and the Oracle Retail system. The common values are decided on between them.

For more information about initial load of the ORGANIZATION_ID cross-reference table, see [Section 7.6, "Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities."](#)

2. Synchronize payment terms using the Oracle AIA process.
3. Synchronize freight terms using the manual process.

7.1.4 Inventory Valuation and Revenue Recognition Accounting Entries Prerequisites

The prerequisites include:

1. Enter these domain value maps (DVMs) using Oracle JDeveloper, and export and set up the following in the PeopleSoft system:
 - a. BUSINESS_-UNIT
 - b. CURRENCY_CODE
2. Create these views in the Oracle Retail database:
 - a. FIF_STG_GL_DATA_VW

- b. IM_FINANCIALS_STG_VW
3. Set up the AIAConfigurationProperties.XML file.

7.1.5 Retail Merchandise Procure to Pay Prerequisites

The prerequisites include:

1. Synchronize suppliers and their locations, payment terms, and currency exchange rates between the Oracle Retail and PeopleSoft applications.
2. Map chart of accounts segments between the Oracle Retail and PeopleSoft applications.
3. Validate chart of accounts code combinations.
4. Set up a calendar for transactions.
5. Import PIP ODI Artifacts (code) and configure connections.

7.1.6 Drill Back and Drill Forward Prerequisites

The prerequisites include:

1. Set up, configure and put into production PeopleSoft GL and Payables and Oracle Retail implementations.
2. Integrate accounting entries and invoice data from Oracle Retail to the PeopleSoft applications.
3. Install and configure Oracle BI Publisher for Oracle Retail.
4. Run the Invoice Oracle Data Integrator (ODI) flow.

7.2 Data Requirements

Data requirements indicate the mandatory data that must be provided to make the integration flows successful.

This section discusses the data requirements of the following process integrations:

- Currency exchange rate
- Payment term
- Supplier information
- Chart of Accounts validation
- Inventory valuation and revenue recognition accounting entries
- Retail Merchandise procure to pay
- Drill back and drill forward

7.2.1 Currency Exchange Rate Data Requirements

No data requirements exist for this process integration.

7.2.2 Payment Term Data Requirements

The process integration for payment term requires:

- In PeopleSoft Payables, payment terms must be assigned to at least one set of books (GL business unit or AP business unit or both) upon creation and generation of ABM.
- Assignment of payment terms to additional set of books (GL business unit or AP business unit or both) generates an ABM for this event.

7.2.2.1 Class Diagram

Figure 7-1, Figure 7-2, and Figure 7-3 are class diagrams for payment term integration.

Figure 7-1 Payment Term Integration class diagram (1 of 3)

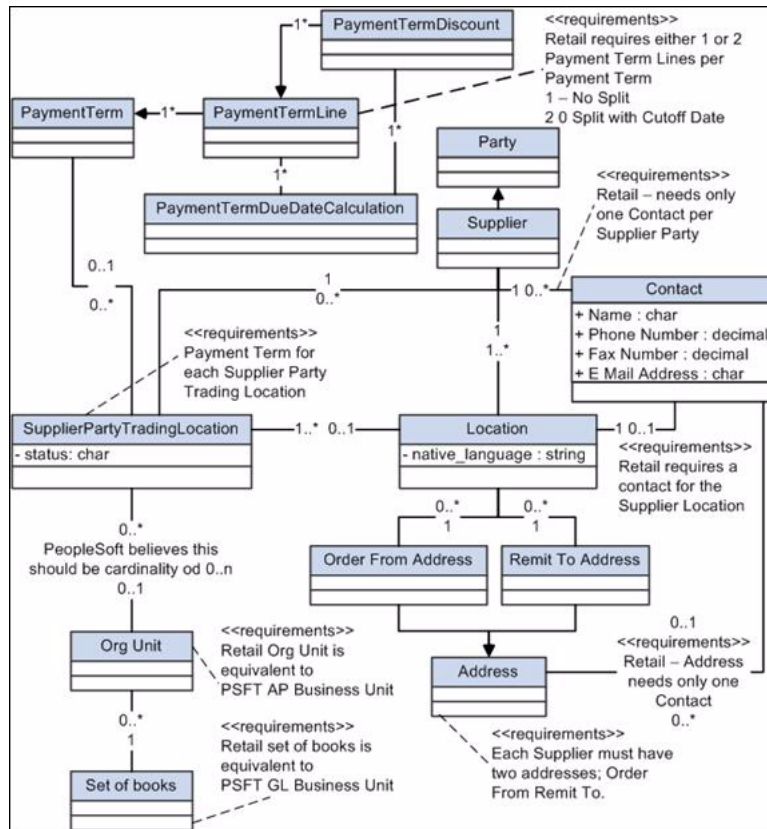


Figure 7-2 Payment Term Integration class diagram (2 of 3)

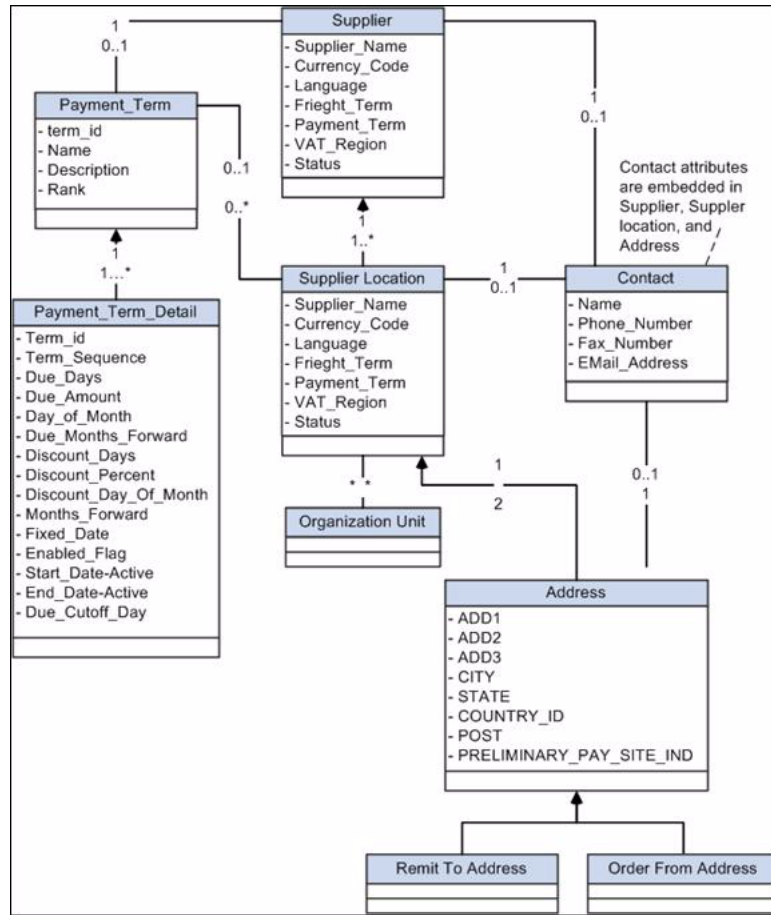
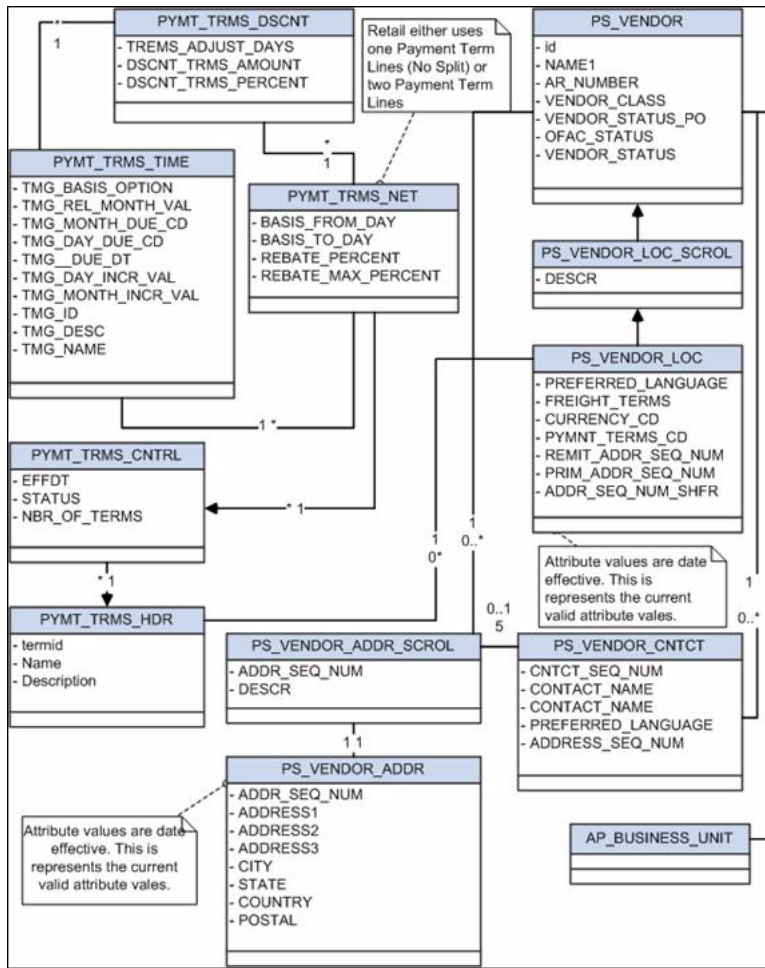


Figure 7-3 Payment Term Integration class diagram (3 of 3)



7.2.3 Supplier Information Data Requirements

The process integration for supplier information requires:

- Suppliers must be assigned to at least one GL business unit (set of books).
- PeopleSoft suppliers must be created with these status attributes:
 - Approved.
 - Classification of supplier.

Other type of vendors, such as attorneys, employees, and HCM, are not synchronized with Oracle Retail.

- The Open for ordering option is selected.

If the supplier is approved but is not open for ordering, the Oracle AIA layer on the Oracle RMS side changes its status to inactive.

- Suppliers must have been assigned at least one Order from address.
- Suppliers must have been assigned at least one Remit to Location address.

7.2.3.1 Class Diagram

Figure 7-4, Figure 7-5, and Figure 7-6 are the class diagrams for supplier integration.

Figure 7-4 Class diagram (1 of 3)

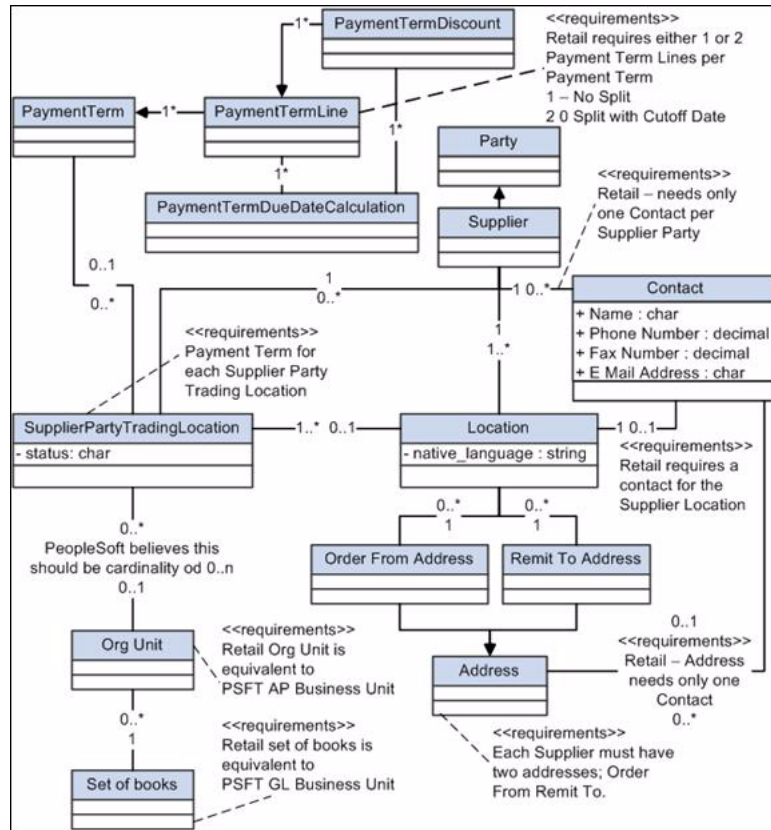


Figure 7-5 Class diagram (2 of 3)

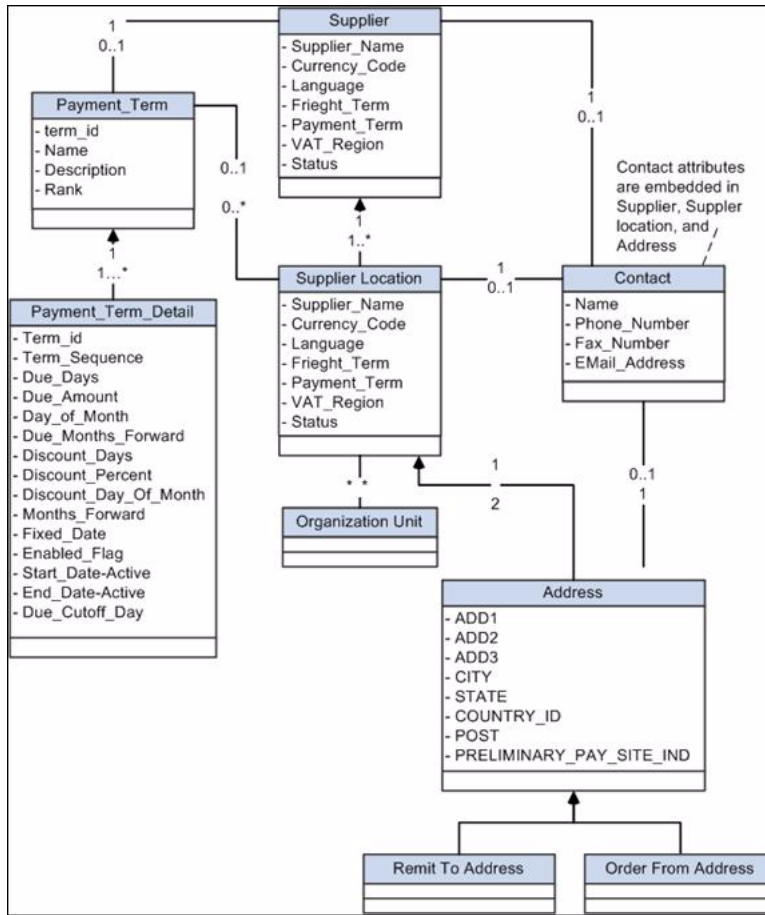
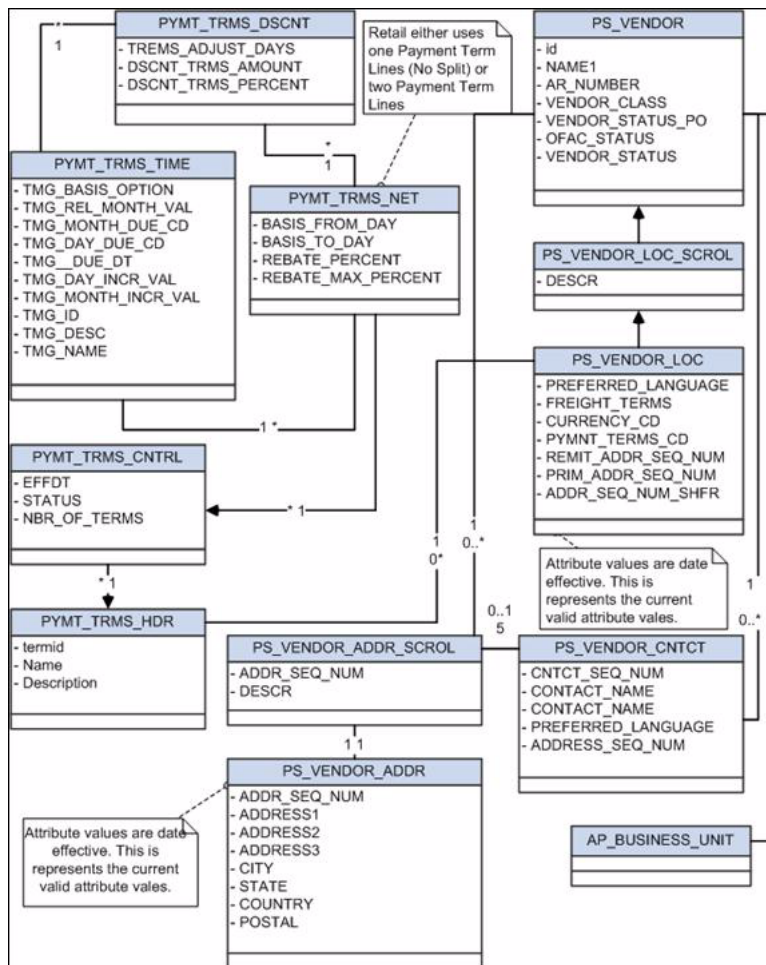


Figure 7-6 Class diagram (3 of 3)



7.2.4 Chart of Accounts Validation Data Requirements

You must manually enter the segment combinations in the GL ACCOUNT MAINTENANCE (Oracle ReSA), GL CROSS REFERENCE (Oracle RMS), and GL CROSS REFERENCE (Oracle ReIM) forms.

7.2.5 Inventory Valuation and Revenue Recognition Accounting Entries Data Requirements

No data requirements exist for this process integration.

7.2.6 Retail Merchandise Procure to Pay Data Requirements

No data requirements exist for this process integration.

7.2.7 Drill Back and Drill Forward Data Requirements

The accounting entry and invoice ODI integration flows were run.

For more information, see [Section 4.1, "Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview"](#) and [Section 5.1, "Process Integration for Retail Merchandise Procure to Pay Overview."](#)

7.3 Setting Up Oracle Retail Applications

Before integration, you must set up Oracle Retail Merchandising System (Oracle RMS), Oracle Retail Invoice Matching (Oracle ReIM), and Oracle Retail Sales Audit (Oracle ReSA).

For more information about setting up Oracle Retail applications, see the *ReIM Operations Guide*, "PeopleSoft Enterprise Financials Integration" and the *RMS Operations Guide*, "AIA Financials Integration."

7.4 Setting Up PeopleSoft GL

To set up PeopleSoft GL, perform these actions:

- Activate service operations, queue, and handlers.
- Activate or create routings.
- Set up end points (connector properties) for provided services.
- Set up recurring run controls for delayed publishing of effective-dated objects.
- Activate validation service operations.
- Set up cross-reference tables.
- Synchronize currency exchange rates.

For more information, see PeopleSoft Enterprise Installation Documentation Update: Oracle Retail Management Integration.

7.5 Setting Up PeopleSoft Payables

To set up PeopleSoft Payables, perform these actions:

- Activate service operations, queue, and handlers.
- Activate or create routings.
- Set up end points (connector properties) for provided services.
- Set up recurring run controls for delayed publishing of effective-dated objects.
- Activate validation service operations.
- Set up cross-reference tables.
- Synchronize payment terms and vendors.

For more information, see PeopleSoft Enterprise Installation Documentation Update: Oracle Retail Management Integration.

7.6 Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities

Before running the supplier flow, you must load the initial ORGANIZATION_ID cross-reference table. Because no automated process for this synchronization is available, you must load this table manually.

This section discusses the following topics:

- [Section 7.6.1, "Populating Cross-References"](#)
- [Section 7.6.2, "Validating Cross-References"](#)

- [Section 7.6.3, "Viewing Cross-Reference Data"](#)

7.6.1 Populating Cross-References

To populate cross-references:

You must manually populate the ORGANIZATION_ID cross-reference table after the install because no process flow exists for this synchronization. Manually add the data into this table using the sql insert statements. The ORGANIZATION_ID table holds the information for business units among the edge applications and the common value.

Sample sql statement:

RETL_01:

```
INSERT INTO "AIA30_XREF"."XREF_DATA" (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES
('orams:/apps/AIAMetaData/xref/ORGANIZATION_ID.xref', 'RETL_01',
'sdfsdklfjere657654', '3333333333333333', 'N', TO_TIMESTAMP('15-FEB-10 10.48.33.',
'DD-MON-RR HH.MI.SS.FF AM'))
```

PSFT_01:

```
INSERT INTO "AIA30_XREF"."XREF_DATA" (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES
('orams:/apps/AIAMetaData/xref/ORGANIZATION_ID.xref', 'PSFT_01',
'sdfsdklfjere657654', 'US01', 'N', TO_TIMESTAMP('15-FEB-10 10.48.33.', 'DD-MON-RR
HH.MI.SS.FF AM'))
```

COMMON:

```
INSERT INTO "AIA30_XREF"."XREF_DATA" (XREF_TABLE_NAME, XREF_COLUMN_NAME, ROW_
NUMBER, VALUE, IS_DELETED, LAST_MODIFIED) VALUES
('orams:/apps/AIAMetaData/xref/ORGANIZATION_ID.xref', 'COMMON',
'sdfsdklfjere657654', '4522145', 'N', TO_TIMESTAMP('15-FEB-10 10.48.33.',
'DD-MON-RR HH.MI.SS.FF AM'))
```

For each set, which includes *RETL_01*, *PSFT_01*, and *COMMON*, the row numbers must be identical to link these records to each other.

Run the script for each business unit that you insert into the cross-reference table. [Example 7-1](#) is an example organization.xml file.

Example 7-1 Sample Organization.xml

```
<xref xmlns="http://xmlns.oracle.com/xref">
<table name="ORGANIZATION_ID">
<columns>
  <column name="RETL_01"/>
  <column name="PSFT_01"/>
</columns>
<rows>
  <row>
    <cell colName="RETL_01">3333333333333333</cell>
    <cell colName="PSFT_01">US001</cell>
  </row>
  <row>
    <cell colName="RETL_01">4444444444444444</cell>
    <cell colName="PSFT_01">US002</cell>
  </row>
</rows>
</table>
</xref>
```

```

<row>
  <cell colName="RETL_01">5555555555555555</cell>
  <cell colName="PSFT_01">CAN01</cell>
</row>
</rows>
</table>
</xref>

```

7.6.2 Validating Cross-References

To validate cross-references:

1. Log in to your database.
2. Query the Table ORGANIZATION_ID to confirm that every organization/business unit used in the XML files has three records.

Use one query from the following:

```

select * from xref_data where xref_table_
name='orands:/apps/AIAMetaData/xref/ORGANIZATION_ID.xref'
select * from xref_data where xref_table_name='ORGANIZATION_ID' and xref_column_
name='PSFT_01'

```

This query retrieves all records in the ORGANIZATION_ID TABLE.

You can modify this query to change the column names to *RETL_01* and *COMMON*.

```
select * from xref_Data
```

```
where row_number in (select row_number from xref_Data where value = 'US01')
```

This query retrieves the three associated records for the org_id you selected.

Replace the value for the business unit you selected. (The number of business units depends on your setup.)

3. Validate each row for which the following three records are written:

RETL_01, *PSFT_01* and *COMMON*

The three records should be linked to each other with a common row number value.

You can export or copy the data from SQL to an Excel spreadsheet. You can sort this data by row number, and so on.

7.6.3 Viewing Cross-Reference Data

To view cross-reference data:

You must sync the data in the cross-reference table with the Business Unit DVM in PeopleSoft. Because *COMMON* is an auto-generated guide and some of this data may not be known to the PeopleSoft customer, data visibility is essential. You must enter this data in the PeopleSoft DVM.

1. Copy and paste from the SQL Worksheet:

If you have access to the database, you also can copy the data into an Excel spreadsheet. From a sql worksheet:

2. Query on the table.

3. Select * from xref_data where xref_table_name='oramds:/apps/AIAMetaData/xref/ORGANIZATION_ID.xref'
4. Right-click in the displayed data, select all/copy, and paste it into an Excel spreadsheet.

7.7 Setting Up PeopleSoft Domain Value Maps

Set up the PeopleSoft DVMs listed in [Figure 7-1](#).

Table 7-1 PeopleSoft DVMs

Name	Description
APBUtoOrgUnitDVM	AP business unit to organization unit
AccountTypes	Account types mapping
BULedgerDVM	Business unit and ledger mapping
CONVERSIONTYPECODE	Rate type mapping
CURRENCY95CODE	Currency code mapping
ChartFieldDVM	ChartField mapping
ChartfieldtoValueSet	ChartField to ValueSet EBM
CountryCodeDVM	Country codes DVM
LanguageCodeDVM	Language code DVM
RetailDefaultCurrency	Default supplier currency
RetailDefaultFreightTerms	Default Retail freight terms
RetailDefaultPayTerms	Default Retail payment terms
StateCodeDVM	State and country table DVM
VendorAddressTypeDVM	Vendor address type DVM
VendorStatusDVM	Vendor status DVM

For more information, see PeopleSoft Enterprise Installation Documentation Update: Oracle Retail Management Integration.

7.8 Manually Synchronizing Freight Terms

Freight is the transportation charge paid by retailers when they receive goods from a supplier. *Freight term* is an agreement between a retailer and a supplier regarding the type and payment of freight.

PeopleSoft Payables is the system of record. Oracle RMS requires the freight terms to generate purchase orders.

You set up PeopleSoft Payables freight term codes and their descriptions manually in Oracle RMS. Incremental changes in freight terms are manually synchronized from PeopleSoft Payables to Oracle RMS; these are not synchronized automatically.

For more information, see the *PeopleSoft Enterprise Application Fundamentals PeopleBook*, "Defining Financials and Supply Chain Management Common Definitions."

7.9 Manually Synchronizing Currency Exchange Rate Types

Oracle Retail allows for multiple currency exchange rate types (such as operational, consolidated, letter of credit or bank, purchase order, customs entry, and transportation) but uses only one type for the default processing—either *Operational* or *Consolidation*.

The PeopleSoft system supports any number of currency exchange rate types and allows for multiple currency exchange rate types. Because Oracle Retail uses only one currency exchange rate for the default processing, the system selects one exchange rate. All of the Oracle Retail products that integrate with PeopleSoft use the selected currency exchange rate.

The PeopleSoft system can have separate currency exchange rate types that are not integrated with Oracle Retail. The relationship between Oracle Retail and PeopleSoft rate types is mapped in a DVM table.

When you manually set up the selected currency exchange rate type in Oracle RMS, Oracle RMS uses the Currency Exchange Type Mapping window to map the external exchange type sent by the PeopleSoft system. Oracle Retail applications use this currency exchange rate type for all transactions.

Because the volume of currency exchange rate types is low, they are maintained and synchronized manually in PeopleSoft GL and Oracle RMS. They are not synchronized automatically. Synchronization is required for the currency exchange rates used by PeopleSoft GL and Oracle RMS.

For more information, see the *PeopleSoft Enterprise Global Options and Reports PeopleBook*, "Processing Multiple Currencies."

7.10 Manually Synchronizing Currency Codes

Oracle Retail and PeopleSoft GL synchronize the currency codes manually. The currency codes mapping between PeopleSoft GL and Oracle Retail is required for the sales audit and stock ledger transactions.

For more information, see the *PeopleSoft Enterprise Components For FSCM PeopleBook*, "Working With Currencies and Market Rates."

7.11 Manually Synchronizing Chart of Accounts Values

A chart of accounts is a listing of all the accounts in a general ledger. You can use a chart of accounts to view specific information about a transaction.

Oracle Retail uses the chart of accounts as reference information to create accounting entries for both sales and inventory transactions. PeopleSoft GL is the system of record for this information. Oracle Retail also stores and uses the GL accounts.

In PeopleSoft, the chart of accounts implies all the accounting ChartFields and values, such as chart of accounts, business unit, and setID. A chart of accounts includes account, department, operating unit, and other ChartFields. The attributes under each ChartField can differ from those of other ChartFields. For example, *Account* can have Account Number, Description, Account Type, Effective Date, and VAT Flag, and *Department* can have Department number, Description, Effective Date, and Manager.

You synchronize charts of accounts manually from PeopleSoft Payables to Oracle Retail. The system does not synchronize them automatically.

The manual setup and synchronization of the chart of accounts reference data includes:

- Initial load of chart of accounts. For a new instance (logical or physical) of Oracle Retail, manually set up the valid chart of accounts segment or ChartFields values for each business unit.
- Incremental chart of accounts synchronization. Manually synchronize new, changed, or deleted Segment or ChartField values from PeopleSoft to Oracle Retail.

The chart of accounts is validated based on the currently effective account code combinations. You must manually synchronize the most current active Chart of Account value row from PeopleSoft GL to Oracle Retail.

The PeopleSoft system can also create future-dated charts of accounts. You must manually update these future-dated charts of accounts when they become effective.

After the chart of accounts values are set up in both systems, each Oracle Retail transaction ready to be transmitted to the PeopleSoft system is validated.

For more information, see the *PeopleSoft Enterprise Application Fundamentals PeopleBook*, "Defining and Using ChartFields."

7.12 Manually Synchronizing Financial Calendars

You manually maintain the calendars in Oracle Retail and PeopleSoft applications; the system does not synchronize them automatically.

For more information, see the *PeopleSoft Enterprise Application Fundamentals PeopleBook*, "Defining Accounting Calendars."

7.13 Mapping Business and Organization Units

A business unit (BU) is an identification code that represents a high-level organization of business information. Use it to define regional or departmental units within a larger organization. A BU maintains its own set of transactions and may have ledgers containing any type of data, such as actual, budget, or forecast information. Transaction tables are keyed by BU. Each BU has a single base currency.

The set of books in Oracle Retail is equivalent to the PeopleSoft GL BU, ledger group, and setID. Many locations, such as warehouses and stores, in Oracle Retail belong to a set of books. The relationship is mapped in an Oracle AIA DVM table.

The Organization Unit in Oracle Retail is equivalent to the PeopleSoft AP BU and setID. You are not required to map setIDs on the Oracle Retail side. The Oracle Retail Organization Units (a group of locations) are mapped with the PeopleSoft Payables BUs.

The Oracle AIA layer maintains the mapping of the BU or setID to the set of books and to the organization unit.

For more information, see the *PeopleSoft Enterprise Payables PeopleBook*, "Setting Up PeopleSoft Payables Business Units" and the *PeopleSoft Enterprise General Ledger PeopleBook*, "Defining Your Operational Structure."

7.14 Working with Cross-References

Cross-references map and connect the records within the application network, and they enable these applications to communicate in the same language. The integration server stores the relationship in a persistent way so that others can refer to it.

For more information about cross-references, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack* and the *Oracle Cross Reference User Guide*.

Table 7–2 lists and describes the Oracle Retail to PeopleSoft Financials PIP cross-references:

Table 7–2 Oracle Retail to PeopleSoft Financials PIP Cross-References

Cross-Reference Table Name	Column Name	Description	Usage
CURRENCYEXCHANGE_ID	COMMON	PeopleSoft GUID	Populate during the currency exchange integration.
RETL_01	Concatenation of Retail FromCurrencyCode:ToCurrencyCode:ConversionType:ConversionDate	--	--
PSFT_01	PeopleSoft GUID	--	--
PAYABLEINVOICE_ID	COMMON	Same GUID as PSFT_01	Populated by the Oracle Data Integrator (ODI) invoice integration and then used by the DrillBackForward web service.
RETL_01	Reference ID from Oracle Retail	--	--
PSFT_01	A GUID generated by the native database	--	--
PAYMENTTERM_ID	COMMON	Generated GUID	Populate during the payment term integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--
PAYMENTTERMLINE_ID	COMMON	Generated GUID	Populate during the payment term integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--
SUPPLIERPARTY_ID	COMMON	Generated GUID	Populate during the supplier integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--
SUPPLIERPARTY_ADDRESS_ID	COMMON	Generated GUID	Populate during the supplier integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--
SUPPLIERPARTY_LOCATION_ID	COMMON	Generated GUID	Populate during the supplier integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--

Table 7–2 (Cont.) Oracle Retail to PeopleSoft Financials PIP Cross-References

Cross-Reference Table Name	Column Name	Description	Usage
ORGANIZATION_ID	COMMON	Generated GUID	Populate for the SupplierParty Integration.
RETL_01	Oracle Retail ID	--	--
PSFT_01	PeopleSoft GUID	--	--

7.15 Working with DVMs

DVMs are a standard feature of the Oracle service-oriented architecture (SOA) suite. They are tables containing mapping between related information in the participating applications. They enable you to equate lookup codes and other static values across applications, for example, *FOOT* and *FT* or *US* and *USA*. These DVM tables are maintained in the Oracle AIA layer. The Oracle AIA layer uses these DVM tables in transforming the messages from one system in the expected format of the other system.

DVM types are seeded for the Oracle Retail to PeopleSoft Financials flows, and administrators can extend the list of mapped values by adding more maps. The DVM data should be synchronized with what the participating applications use. This synchronization should occur before any initial loads are run or any incremental transactional flows are initiated.

Note: For any DVM that is not pre-seeded, the first row has description information, such as *Enter Org Unit here*. Remove this first row because it causes issues in the Oracle Data Integrator flow.

Table 7–3 lists and describes the DVMs for the Oracle Retail to PeopleSoft Financials PIP:

Table 7–3 DVMs for the Oracle Retail to PeopleSoft Financials PIP

DVM Type	DVM Column Name	Comments
BUSINESS_UNIT	COMMON, RETL_01, PSFT_01	Maps the Oracle Retail Set Of Books IDs to PeopleSoft GL business units.
CHARTOFACCOUNTS_ACCOUNTSTATUS	COMMON, RETL_01, PSFT_01	A hard-coded DVM that should not be changed. Maps the PeopleSoft and EBO value of true or false to the Oracle Retail value of valid or invalid.
CHARTOFACCOUNTS_GLELEMENT	COMMON, RETL_01, PSFT_01	Maps Oracle Retail segment values to PeopleSoft GL elements.
ADDRESS_COUNTRYID	COMMON, RETL_01, PSFT_01	Maps the country codes between Oracle Retail and the PeopleSoft system.
CURRENCY_CODE	COMMON, RETL_01, PSFT_01	Maps the currency codes between Oracle Retail and the PeopleSoft system.
CURRENCYEXCHANGE_CONVERSIONTYPECODE	COMMON, RETL_01, PSFT_01	Maps the currency exchange codes between Oracle Retail and the PeopleSoft system.
LANGUAGE_CODE	COMMON, RETL_01, PSFT_01	Maps the language code between Oracle Retail and the PeopleSoft system. The system uses this DVM to determine which language to use to send the translatable fields to Oracle Retail.

Table 7–3 (Cont.) DVMs for the Oracle Retail to PeopleSoft Financials PIP

DVM Type	DVM Column Name	Comments
ORACLERETAILSEGMENTC OLNAMES	CHARTOFACCOUNTS_ GLELEMENT, STG_FIF_GL_ DATA, IM_FINANCIALS_ STG, IM_AP_STAGE_DETAIL	A hard-coded DVM that should not be changed. Lists the column names for all 20 segments for the tables in the Oracle RMS, Oracle ReIM, and Oracle ReIM Journal Entry systems. The system uses this for the accounting entry ODI flow.
STATE	COMMON, RETL_01, PSFT_ 01	Maps the state codes between Oracle Retail and PeopleSoft Payables.
SUPPLIERPARTY_ ADDRESSTYPE	COMMON, RETL_01, PSFT_ 01	Maps the supplier's address type between Oracle Retail and PeopleSoft Payables.
SUPPLIERPARTY_ STATUSCODE	COMMON, RETL_01, PSFT_ 01	Maps the supplier's status between Oracle Retail and PeopleSoft Payables.

7.16 Handling Errors

Based on the roles defined for the services, the system sends email notifications if a service errors out. [Table 7–4](#) lists the errors generated by the Oracle Retail to PeopleSoft Financials PIP services:

Table 7–4 Errors Generated by the Oracle Retail to PeopleSoft Financials PIP

Error Code	Message Text
AIA_ERR_AIARTL2PSFT_1001	Invalid Requesting System; Requesting Systems must be Oracle ReSA, Oracle ReIM, or Oracle RMS
AIA_ERR_AIARTL2PSFT_1002	Invalid Set Of Books ID; Set Of Books ID must in BUSINESS_UNIT DVM
AIA_ERR_AIARTL2PSFT_1003	Invalid Currency Conversion Type Code

You can set the Integration Broker process to handle multiple processes so that if an error is generated, the process can continue to complete the remaining transactions.

For more information about the errors generated by Oracle Retail or PeopleSoft applications, see that product's documentation.

For more information about Oracle AIA error handling, see *Oracle Fusion Middleware Infrastructure Components and Utilities User's Guide for Oracle Application Integration Architecture Foundation Pack*, "Setting Up and Using Error Handling and Logging."

7.17 Viewing EIMs

For more information about how services are mapped, see the My Oracle Support document: EBO Implementation Maps (EIMs) 881022.1.

7.18 Setting Configuration Properties

Set these properties in the AIAConfigurationProperties.XML file. This file is available in <aia_instances>/config/.

For more information about requirements for working with AIAConfigurationProperties.xml, see *Oracle Fusion Middleware Developer's Guide for*

Oracle Application Integration Architecture Foundation Pack, "Building AIA Integration Flows," How to Set Up AIA Workstation."

For more information, see *Oracle Fusion Middleware Developer's Guide for Oracle Application Integration Architecture Foundation Pack*, "Configuring and Using Oracle Enterprise Repository as the Oracle AIA SOA Repository."

Table 7–5 describes the settings for the ProcessGLAccountValidationPeopleSoftProvABCServiceImpl service property:

Table 7–5 ProcessGLAccountValidationPeopleSoftProvABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code, defined in Oracle Enterprise Repository (OER) from the account validation provider.
Routing.GL_CHARTFIELD.PSFT_01.EndpointURI	http://\${psftfscm.http.host}:\${psftfscm.http.port}/PSIGW/PeopleSoftServiceListeningConnector/\${psftfscm.http.default.local.node}	PeopleSoft GL_Chartfield web service endpoint location. A SOAP endpoint URL.
Routing.GL_CHARTFIELD.CAVS.EndpointURI	http://\${http.host.name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	Composite Application Validation System (CAVS) endpoint location to simulate a sync response for the PeopleSoft GL_Chartfield web service.
Routing.GL_CHARTFIELD.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–6 describes settings for the SyncCurrencyExchangeListPeopleSoftReqABCServiceImpl service property:

Table 7–6 SyncCurrencyExchangeListPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in OER) from which messages originate.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–6 (Cont.) SyncCurrencyExchangeListPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
ABCSExtension.PostProcessE BM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessA BM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.SyncCurrencyExchan geListPeopleSoftReqABCSSim pl.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSyst emServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.SyncCurrencyExchan geListPeopleSoftReqABCSSim pl.MessageProcessingInstructi on.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.

Table 7–7 describes settings for the SyncPaymentTermListPeopleSoftReqABCServiceImpl service property:

Table 7–7 SyncPaymentTermListPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in OER) from which messages originate.
Routing.PaymentTermEBS.Ro uteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessAB M	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEB M	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessE BM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessA BM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.PaymentTermEBS.C AVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSyst emServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.PaymentTermEBS.M essageProcessingInstruction.E nvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which can be PRODUCTION or CAVS.

Table 7–8 describes settings for the CreatePaymentTermPeopleSoftReqABCServiceImpl service property:

Table 7–8 CreatePaymentTermPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in OER) from which messages originate.
Routing.PaymentTermEBS.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.PaymentTermEBS.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.PaymentTermEBS.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which can be PRODUCTION or CAVS.

[Table 7–9](#) describes settings for the UpdatePaymentTermPeopleSoftReqABCServiceImpl service property:

Table 7–9 UpdatePaymentTermPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) from which messages originate.
Routing.PaymentTermEBS.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.PaymentTermEBS.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.PaymentTermEBS.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.

[Table 7–10](#) describes settings for the `CreateSupplierPartyPeopleSoftReqABCImpl` service property:

Table 7–10 CreateSupplierPartyPeopleSoftReqABCImpl Service Property

Property Name	Value/Default Values	Description
Sender.SystemID	PSFT_01	PeopleSoft system instance code (defined in OER) from which messages originate.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.CreateSupplierPartyPeopleSoftReqABCImpl.CAVS.EndpointURI	http://\${http:host name}:\${http:port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.CreateSupplierPartyPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.

[Table 7–11](#) describes settings for the `UpdateSupplierPartyPeopleSoftReqABCImpl` service property:

Table 7–11 UpdateSupplierPartyPeopleSoftReqABCImpl Service Property

Property Name	Value/Default Values	Description
Sender.SystemID	PSFT_01	PeopleSoft system instance code (defined in OER) from which messages originate.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–11 (Cont.) UpdateSupplierPartyPeopleSoftReqABCServiceImpl Service Property

Property Name	Value/Default Values	Description
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.UpdateSupplierPartyPeopleSoftReqABCServiceImpl.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.UpdateSupplierPartyPeopleSoftReqABCServiceImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.

[Table 7–12](#) describes settings for the SyncCurrencyExchangeListRetailProvABCServiceImpl service property:

Table 7–12 SyncCurrencyExchangeListRetailProvABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the currency exchange provider.
Routing.SyncCurrencyExchangeListRetailProvJMSProducer.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SyncCurrencyExchangeListRetailProvJMSProducer.RETL_01.EndpointURI	http://\${http.host name}:\${http.port}/orabpel/default/SyncCurrencyExchangeListRetailProvJMSProducer/1.0	Oracle Retail currency exchange endpoint wsdl location. This URL normally resides on the SOA server.
Routing.SyncCurrencyExchangeListRetailProvJMSProducer.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/asyncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail currency exchange endpoint wsdl location.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

[Table 7–13](#) describes settings for the SyncPaymentTermListRetailProvABCServiceImpl service property:

Table 7–13 SyncPaymentTermListRetailProvABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the payment term provider.
Routing.PayTermService.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.PayTermService.RETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/PayTermBean/PayTermService	Oracle Retail PayTermService web service endpoint location. A SOAP endpoint URL.
Routing.PayTermService.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail PayTermService endpoint wsdl location.
Retail.Language	1	Language of the Oracle Retail system. This value corresponds with the LANGUAGE_CODE DVM table.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

[Table 7–14](#) describes settings for the UpdatePaymentTermRetailProvABCServiceImpl service property:

Table 7–14 UpdatePaymentTermRetailProvABCServiceImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the payment term provider.
Routing.PayTermService.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.PayTermService.RETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/PayTermBean/PayTermService	Oracle Retail PayTermService web service endpoint location. A SOAP endpoint URL.
Routing.PayTermService.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail PayTermService endpoint wsdl location.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–14 (Cont.) UpdatePaymentTermRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–15 describes settings for the CreatePaymentTermRetailProvABCImpl service property:

Table 7–15 CreatePaymentTermRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the payment term provider.
Routing.PayTermService.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.PayTermService.RETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/PayTermBean/PayTermService	Oracle Retail PayTermService web service endpoint location. A SOAP endpoint URL.
Routing.PayTermService.CAVS.EndpointURI	http://\${http.host}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail PayTermService endpoint wsdl location.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–16 describes settings for the SyncSupplierPartyListRetailProvABCImpl service property:

Table 7–16 SyncSupplierPartyListRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the supplier party provider.
Routing.SupplierServicePort.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SupplierServicePortRETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/SupplierBean/SupplierService	Oracle Retail SupplierParty web service endpoint location. A SOAP endpoint URL.

Table 7–16 (Cont.) SyncSupplierPartyListRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Routing.SupplierServicePort.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator.	CAVS endpoint location to simulate a sync response for the Oracle Retail SupplierParty service endpoint wsdl location.
Retail.Language	1	Language of the Oracle Retail system. This value corresponds with the LANGUAGE_CODE DVM table.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Retail.ActiveCode	A	Used to determine if the inbound supplier is active or inactive, after which the corresponding value is passed to Oracle Retail.
Retail.InactiveCode	I	Used to determine if the inbound supplier is active or inactive, after which the corresponding value is passed to Oracle Retail.

Table 7–17 describes settings for the CreateSupplierPartyRetailProvABCImpl service property:

Table 7–17 CreateSupplierPartyRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the supplier party provider.
Routing.SupplierServicePort.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SupplierServicePort.RETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/SupplierBean/SupplierService	Oracle Retail SupplierParty web service endpoint location. A SOAP endpoint URL.
Routing.SupplierServicePort.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail SupplierParty service endpoint wsdl location.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–18 describes settings for the SyncSupplierPartyListRetailProvABCImpl service property:

Table 7–18 SyncSupplierPartyListRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	Oracle Retail system instance code (defined in OER) from the supplier party provider.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SyncSupplierPartyListPeopleSoftReqABCImpl.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.SyncSupplierPartyListPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the verification system.

Table 7–19 describes settings for the UpdateSupplierPartyRetailProvABCImpl service property:

Table 7–19 UpdateSupplierPartyRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from the supplier party provider.
Routing.SupplierServicePort.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SupplierServicePort.RETL_01.EndpointURI	http://\${retail.rsl.host}:\${retail.rsl.port}/SupplierBean/SupplierService	Oracle Retail SupplierParty web service endpoint location. A SOAP endpoint URL.
Routing.SupplierServicePort.CAVS.EndpointURI	http://\${http.host name}:\${http.port}/AIAValidationSystemServlet/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail SupplierParty service endpoint wsdl location.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–19 (Cont.) UpdateSupplierPartyRetailProvABCImpl Service Property

Property Name	Value/Default Values	Description
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.

Table 7–20 describes settings for the ProcessGLAccountValidationRetailReqABCImpl service property:

Table 7–20 ProcessGLAccountValidationRetailReqABCImpl Service Property

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in OER) from which messages originate.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
ABCSExtension.PreProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PreProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessEBM	true/false Default = false	Controls whether to route messages to the extensibility system.
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.ProcessGLAccountValidationRetailReqABCImpl.CAVS.EndpointURI	http://\${http.host}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI when CAVS is enabled.
Routing.ProcessGLAccountValidationRetailReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	Code that identifies the environment the ABCS is running in, which could be PRODUCTION or CAVS.

Table 7–21 describes settings for the DrillBackForward service property:

Table 7–21 DrillBackForward Service Property

Property Name	Value/Default Values	Description
PSFTUrl.ProtocolHostPort	http://\${psftfscm.http.host}:\${psftfscm.http.port}	The protocol, host, and port of the PeopleSoft service location.
PSFTUrl.Servlet	\${psftfscm.application.url.servlet}	The name of the PeopleSoft FSCM Application servlet.
PSFTUrl.Site	\${psftfscm.application.url.site}	The name of the PeopleSoft FSCM Application site.

Table 7–21 (Cont.) DrillBackForward Service Property

Property Name	Value/Default Values	Description
PSFTUrl.PortalNodeContent	EMPLOYEE/ERP/c	The Port, Node, and Content for the PeopleSoft FSCM application. The format of this information is: <portal>/<node>/<content>
PSFT.Invoice.WS.Location	http://\${psftfscm.http.host}:\${psftfscm.http.port}/PSIGW/PeopleSoftServiceListeningConnector/\${psftfscm.http.default.local.node}	PeopleSoft Invoice web service endpoint location. A SOAP endpoint URL.
RETAIL.URL.WS.Location	http://\${retail.rsl.host}:\${retail.rsl.port}/ReportLocatorBean/ReportLocatorService	Oracle Retail Report Locator service endpoint location. A SOAP endpoint URL.

