

Oracle® Application Integration Architecture

Oracle Retail Merchandising Integration Pack for PeopleSoft
Enterprise Financials Implementation Guide

Release 2.5

E15778-04

March 2013

The Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials provides integration to a robust enterprise financial system to complement the Oracle Retail Merchandising system in a retail customer environment.

E15778-04

Copyright © 2001, 2013, Oracle and/or its affiliates. All rights reserved.

Primary Author: Oracle Corporation

Contributing Author:

Contributor:

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

Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials 2.5 – Implementation Guide Preface	4
Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide	4
Oracle Application Integration Architecture Foundation Pack Concepts and Technologies Guide	4
Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide	5
Oracle Application Integration Architecture Process Integration Packs.....	6
Additional Resources	6
Chapter 1: Understanding the Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials 2.5	7
Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials Overview	7
Key Benefits	10
Participating Applications Overview	10
Retail Sales Financial Business Process Flow	12
Retail Inventory Financial Business Process Flow.....	14
Retail Merchandise Procure to Pay Business Process Flow	16
Solution Assumptions and Constraints	19
Chapter 2: Life Cycle Management for Reference Data.....	21
Currency Exchange Rate Integration	21
Payment Term Integration.....	37
PayTermService	55
Suppliers Information Integration	56
Chapter 3: Chart of Accounts Validation Integration Flow.....	77
Requesting Chart of Accounts Combination Validation	77
Oracle Retail Interfaces	80
PeopleSoft GL Interfaces	80
Data Requirements	81
Core AIA Components	81
Integration Services.....	81

Chapter 4: Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries	84
Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview.....	84
Performing Setup Tasks.....	86
Configuring and Generating Data in Oracle Retail.....	87
Picking Up and Transforming the Data	88
Configuring and Running the Process Integration for Accounting Entries.....	90
Chapter 5: Process Integration for Retail Merchandise Procure to Pay	95
Process Integration for Retail Merchandise Procure to Pay Overview	95
Performing Setup Tasks.....	98
Configuring and Generating Data in Oracle Retail.....	99
Picking Up and Transforming the Data	100
Configuring and Running the Process Integration for Retail Merchandise Procure to Pay ..	102
Chapter 6: Drill Back and Drill Forward	107
Drill Back and Drill Forward Overview.....	107
Drill Back and Drill Forward Details.....	109
Service Solution Components	111
Data Requirements	115
Chapter 7: Implementing Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials	116
Setting Up Oracle Retail Applications	116
Setting Up PeopleSoft GL	117
Setting Up PeopleSoft Payables	117
Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities	117
Setting Up PeopleSoft Domain Value Maps	123
Manual Synchronization of Freight Terms	124
Manual Synchronization of Currency Exchange Rate Types.....	124
Manual Synchronization of Currency Codes.....	124
Manual Synchronization of Chart of Accounts Values	125
Manual Synchronization of Financial Calendar.....	125
Mapping of Business and Organization Units	126
Working with Cross-References.....	126
Working with DVMs	128
Handling Errors	130
Viewing EBO Implementation Maps (EIMs)	130

Setting Configuration Properties 130

Index 141

Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials 2.5 – Implementation Guide Preface

This preface discusses:

- Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide
- The Oracle Application Integration Architecture - Foundation Pack: Concepts and Technologies Guide
- Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide
- Oracle Application Integration Architecture Process Integration Packs
- Additional resources

Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide

The *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide* provides conceptual, setup, and usage information for the following Core Infrastructure Components:

- The Business Service Repository (BSR).
- The Composite Application Validation System (CAVS).
- Error handling and logging.
- The Diagnostics Framework.

Oracle Application Integration Architecture Foundation Pack Concepts and Technologies Guide

The *Oracle Application Integration Architecture - Foundation Pack: Concepts and Technologies Guide* is a companion volume to the *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide* and *Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide*. The *Oracle Application Integration Architecture - Foundation Pack: Concepts and Technologies Guide* provides definitions of fundamental Oracle Application Integration Architecture (AIA) concepts and discusses:

- Oracle AIA.
- Enterprise business objects and enterprise business messages.
- Enterprise business services.
- Application business connector services.
- Interaction patterns.
- Extensibility.
- Versioning.
- Business processes.
- Batch processing.
- Infrastructure services.
- Security

Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide

The *Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide* is a companion volume to *Oracle Application Integration Architecture - Foundation Pack: Concepts and Technologies Guide* and *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*.

The *Oracle Application Integration Architecture - Foundation Pack: Integration Developer's Guide* discusses how to:

- Create an integration scenario.
- Define business service patterns.
- Design and develop enterprise business services.
- Design and develop enterprise business flows.
- Design and construct application business connector services.
- Work with message transformation, enrichment, and configuration.
- Develop custom xpath functions.
- Design and construct JMS Adapter services.
- Work with enterprise message headers.
- Work with message routing.
- Work with transactions.
- Develop Oracle AIA services to work with the Composite Application Validation System (CAVS).

- Configure Oracle AIA processes to be eligible for error handling and logging.
- Extend enterprise business objects.

In addition, this book provides, Application Integration Architecture naming standards.

Oracle Application Integration Architecture Process Integration Packs

A process integration pack (PIP) is a prebuilt set of integrated orchestration flows, application integration logic, and extensible enterprise business objects and services required to manage the state and execution of a defined set of activities or tasks between specific Oracle applications associated with a given process. A PIP provides everything you need to deploy a selected integrated business process area. The PIP product offering is suited to those customers seeking to rapidly implement a discreet business process.

Additional Resources

The following resources are available:

Resource	Location
Installation Guide	My Oracle Support https://metalink.oracle.com/
Documentation updates	My Oracle Support https://metalink.oracle.com/
Release Notes	Oracle Technology Network http://www.oracle.com/technology/
Known issues, workarounds, and current list of patches	My Oracle Support https://metalink.oracle.com/

Chapter 1: Understanding the Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials 2.5

This chapter provides an overview of the Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials and discusses:

- Key benefits
- Participating applications
- Retail Sales Financial business process flow
- Retail Inventory Financial business process flow
- Retail Procure to Pay business process flow
- Solution assumptions and constraints

Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials Overview

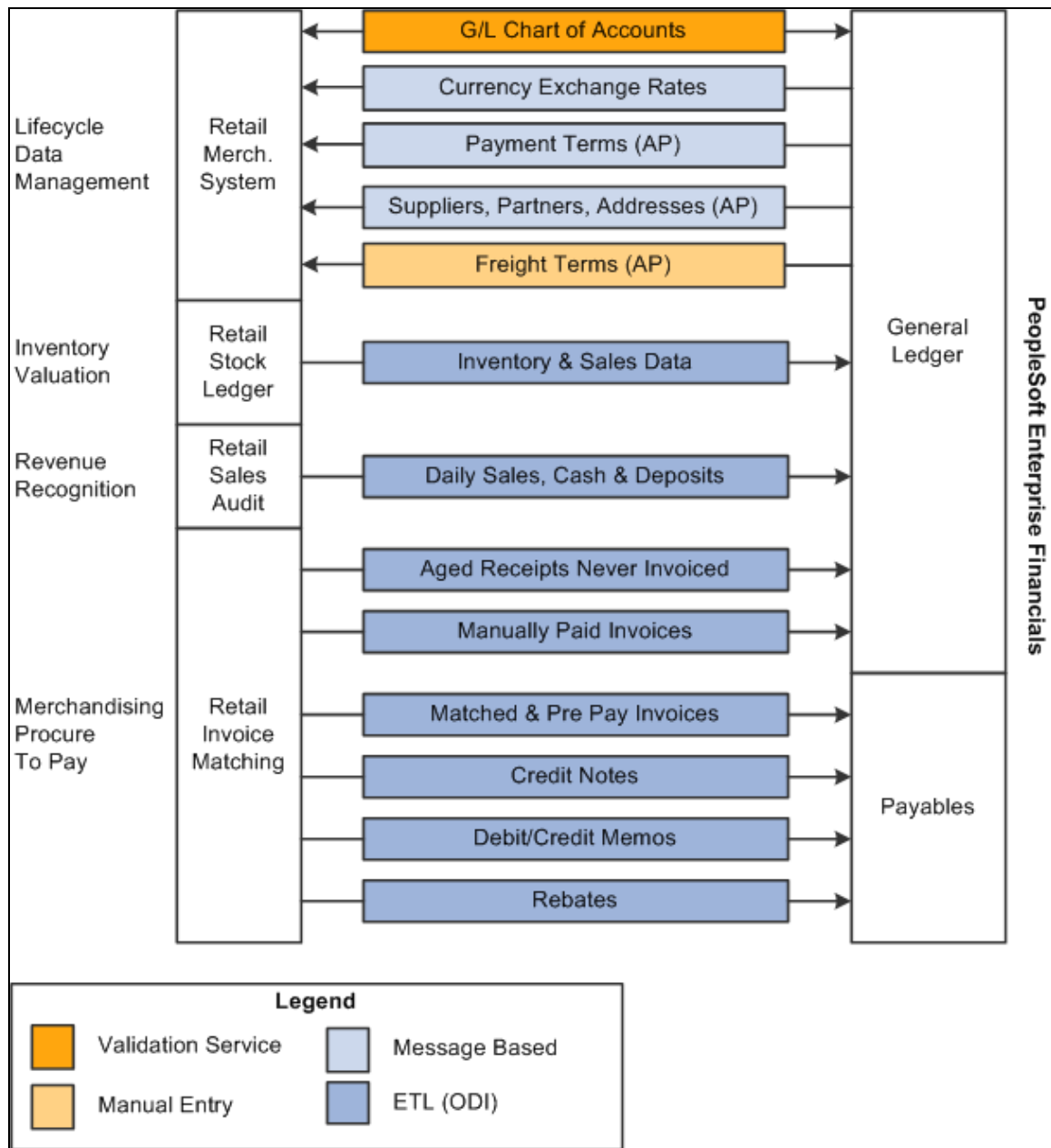
The Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials provides integration to a robust enterprise financial system to complement the Oracle Retail Merchandising system in a retail customer environment.

This Process Integration Pack (PIP) includes these four processes:

- **Life Cycle Data Management:** This process provides the data synchronization for the initial load prior to implementation and incremental data creation and maintenance after implementation. This process synchronizes suppliers, payment terms, and currency exchange rates from the PeopleSoft applications to the Oracle Retail Merchandising System (RMS). PeopleSoft Payables is the source of suppliers, payment terms, and freight terms. Because the freight terms are static in nature and their volume is very low, they are synchronized between the two systems manually. This process enables users of the Oracle Retail Merchandise suite to carry out functions with data that is shared with the PeopleSoft applications throughout the life cycle of creations and updates of the data.
- **Inventory Valuation (Retail stock ledger):** This process enables the posting of accounting entries generated from transactions that change the value of sellable products at a retailer's inventory locations (stores and warehouses) to the appropriate ledgers from Oracle Retail Merchandising – stock ledger to PeopleSoft General Ledger (GL). This process records the financial impact of changes in the sellable inventory in store and warehouse locations. Valuation of sellable inventory in the 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 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 the posting of accounting entries generated from sales and returns transactions from the retailer's stores for revenue and cash reconciliation to the appropriate ledgers. In this process, the data flows from Oracle Retail Sales Audit (ReSA) to PeopleSoft GL. This process records the financial impact of sale/return, cash reconciliation, and void transactions from stores. The Revenue Recognition process begins when store transactions (sales and returns) are processed by ReSA. For each store transaction, 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 (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 trigger the payment authorization of the supplier's invoice. Invoices may be authorized for payment prior to 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 ReIM to PeopleSoft Payables and GL. Other accounting transactions are generated from ReIM to write off aged receipts that were never invoiced and to post accounting distribution for manually paid or prepaid invoices after receipt.

This diagram illustrates the Oracle Retail to PeopleSoft Financials process flows:



Oracle Retail to PeopleSoft Financials PIP process flow

This PIP does not synchronize chart of accounts from PeopleSoft GL to Oracle Retail. Chart 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 impacts in Oracle Retail, the code combinations are validated by a service provided by the PeopleSoft GL. This 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 and Oracle Retail users to access information about the financial transactions from the integrated systems in a seamless and intuitive manner. It provides the users an end-to-end traceability between PeopleSoft Payables and General Ledger back to RMS, ReSA, and ReIM. From PeopleSoft Enterprise Financials, users can drill back to query data from the Oracle Retail applications. The drill back is initiated from the PeopleSoft Financials User Interfaces (UI). Oracle Retail application users can drill forward to query integrated data from PeopleSoft Enterprise Financials. This drill back and drill forward feature provides traceability from retail transactions to journal entries and from invoices to vouchers and, for both, back between Oracle Retail and PeopleSoft. This enables processes to support supplier servicing and financial auditing.

Key Benefits

Here are the key benefits of this PIP:

- This integration is not a point-to-point integration between the PeopleSoft and Oracle Retail applications. This PIP implementation is independent of version of integrated applications. An Application Integration Architecture (AIA) layer serves as an intermediate thin layer of application between PeopleSoft and Oracle Retail. This integration remains synchronized with the new releases of the edge applications.
- Audited transaction data is exported to the PeopleSoft Financial applications days before the traditional audit process permits. The Financials applications can use this timely data in a proactive manner, which results in increased productivity and operational efficiencies.
- Total cost of ownership for Oracle and its customers is reduced.
- Seamless drill back and drill forward actions between the PeopleSoft and Oracle Retail applications. The Oracle Retail as well as PeopleSoft users do not require any training on inquiring the specific data in either application.

Participating Applications Overview

This section provides an overview of these applications participating in the process integration:

- Oracle Retail Merchandising System
- Oracle Retail Sales Audit
- Oracle Retail Invoice Matching
- PeopleSoft Payables
- PeopleSoft General Ledger

Oracle Retail Merchandising System

Oracle Retail Merchandising (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, RMS provides retailers with a complete end-to-end solution and is the most comprehensive and integrated solution for global retailing.

For more information, see the Oracle Retail Merchandising System User Guide.

Oracle Retail Sales Audit

Oracle Retail Sales Audit (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, ReSA enables retailers to make better decisions to improve merchandise operations and transform the economics of their business.

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

Oracle Retail Invoice Matching

Oracle Retail Invoice Matching (ReIM) is a market-leading solution for retailers who need an automated application to better manage reconciliation and payment of purchase orders. 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 right internal teams for resolution and compliance controls.

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

PeopleSoft Payables

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 reducing cycle times and errors. Other features include a flexible, user-defined system setup, extensive vendor maintenance, digital signatures, financials sanction validation, and powerful inquiry and analytical capabilities.

For more information, see the PeopleSoft Enterprise Payables 9.0 PeopleBook.

PeopleSoft General Ledger

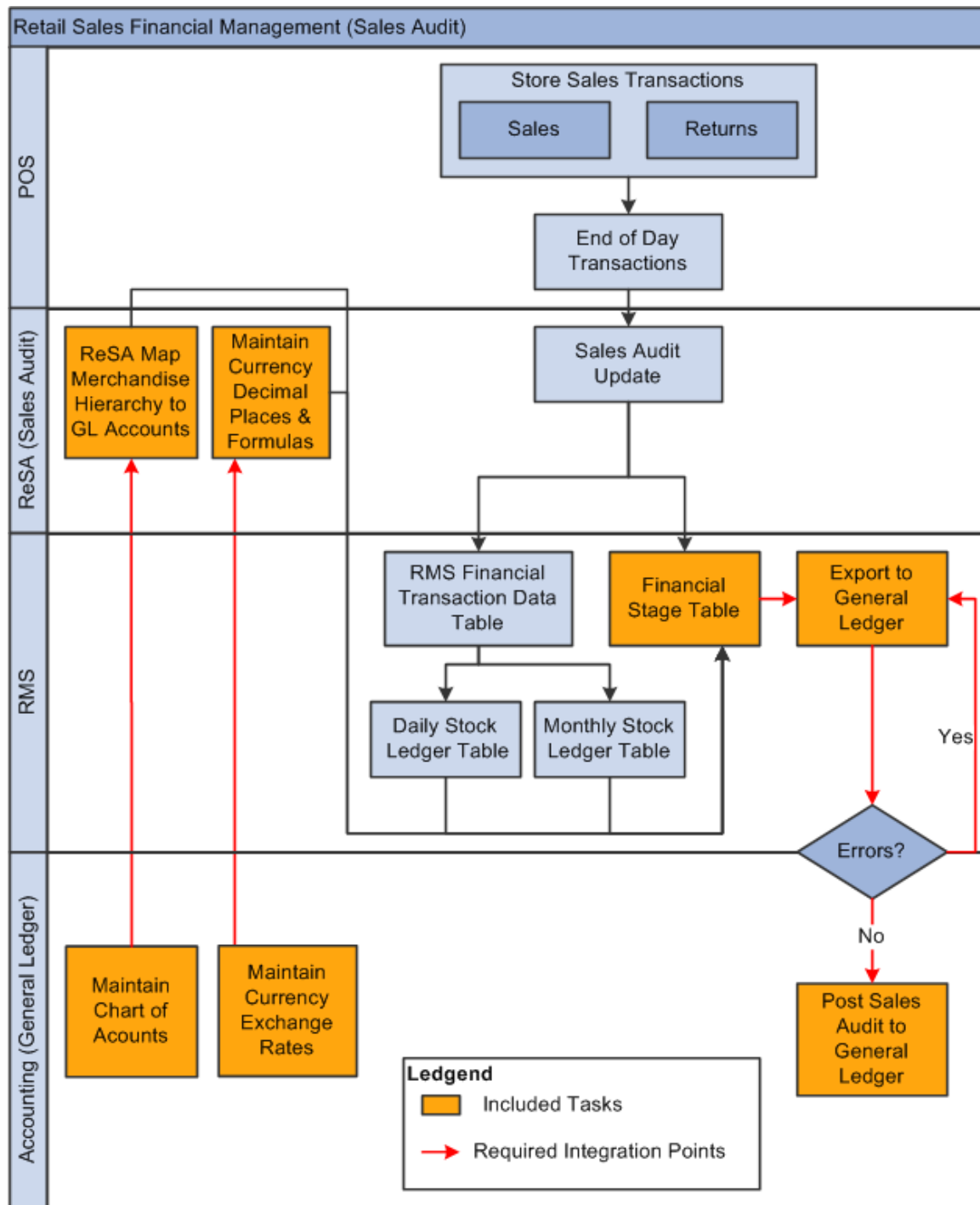
PeopleSoft General Ledger (GL) offers a fully automated close and consolidation solution for legal and management reporting, including support for Generally Accepted Accounting Principles (GAAP) and International Financial Reporting Standards (IFRS). Transactions are automatically processed and validated 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, see the PeopleSoft Enterprise General Ledger 9.0 PeopleBook

Retail Sales Financial Business Process Flow

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

This diagram illustrates the Retail Sales Financial business process flow:



Retail Sales Financial business process flow

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

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 ReSA is also posted directly to the 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 ReSA) and the Currency Exchange Rates.

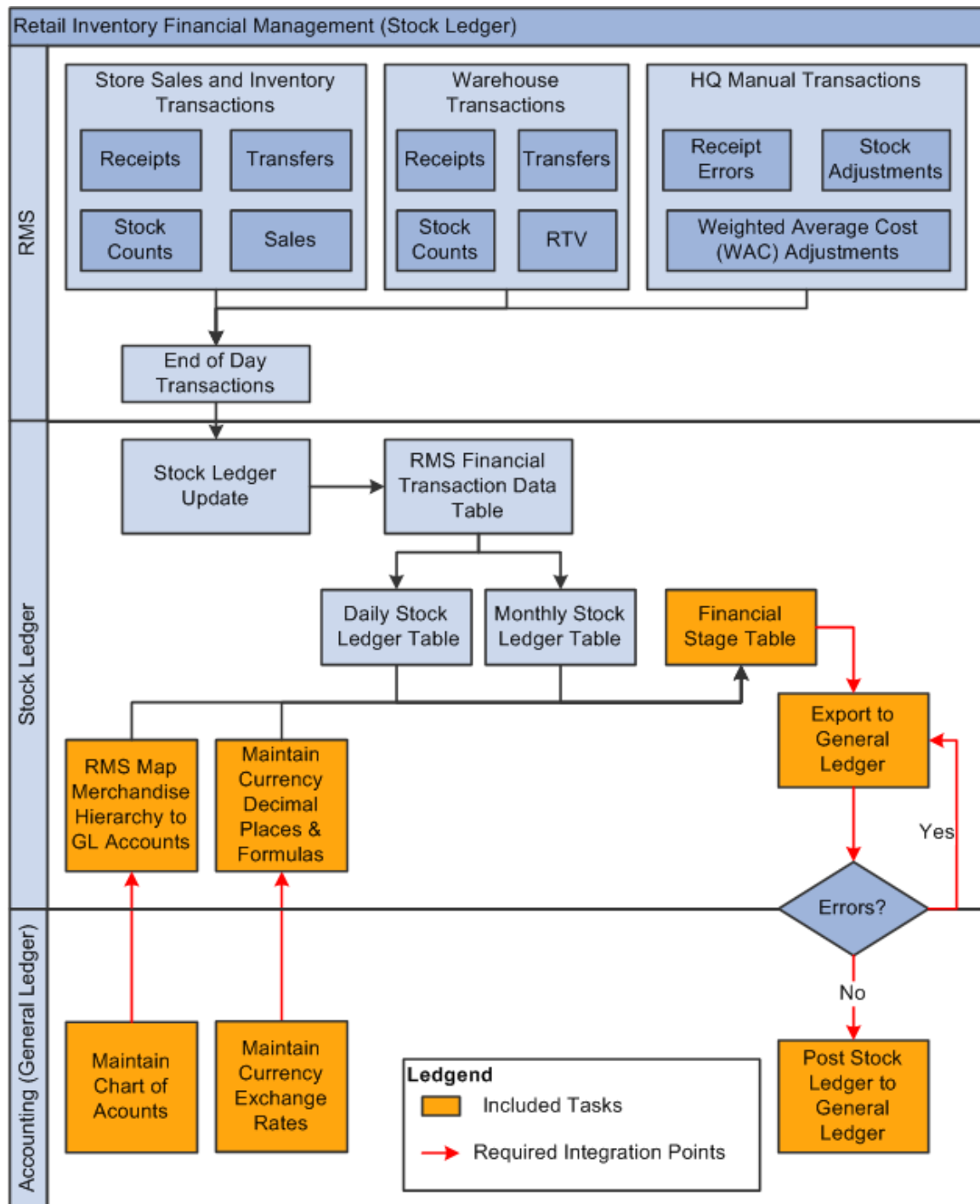
The PeopleSoft Journal Generator loads the accounting entries from Oracle Retail to PeopleSoft GL journal entries. The accounting tables are referenced on the accounting entry definition defined for each type of accounting entry transaction. The Journal Generator uses the new accounting entry definitions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

Retail Inventory Financial Business Process Flow

The Retail Inventory Financial business process consists of these integration flows:

- Post stock ledger from RMS to PeopleSoft GL.
- Post write-offs (aged receipts, not invoiced to ledger) from RMS to PeopleSoft GL.

This diagram illustrates the Retail Inventory Financial business process flow:



Oracle Retail Inventory Financials business process flow

The stock ledger in RMS records the financial results of the merchandising processes that occur in the Retail system, such as buying, selling, price changes, transfers, and so on. All of these transactions are recorded in the 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. 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 through the use of table triggers or batch, by means of the appropriate General Ledger account combinations (maintained in the RMS cross-reference table in Oracle Retail) and the currency exchange rates.

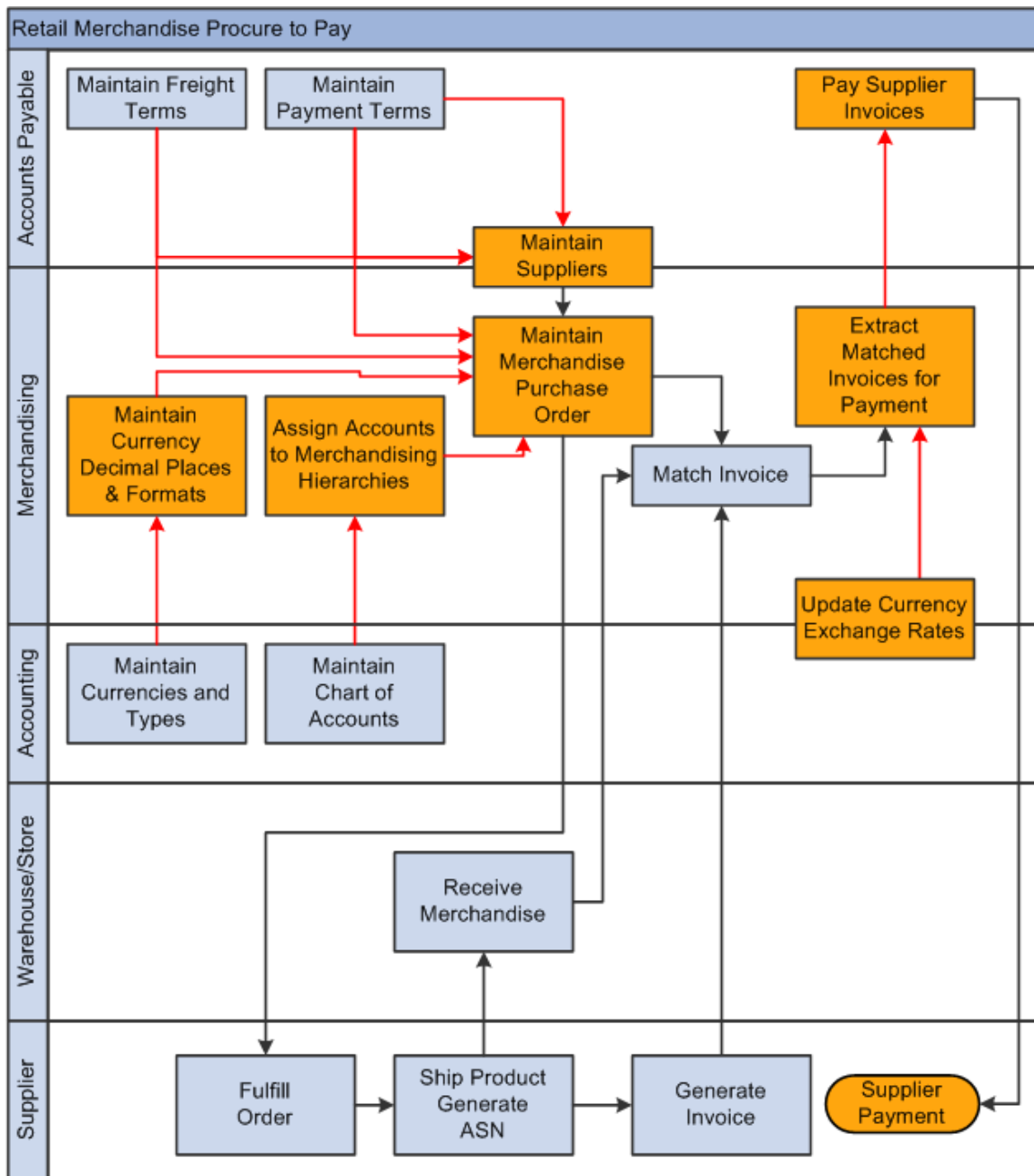
The PeopleSoft Journal Generator loads the accounting entries from Oracle Retail and creates PeopleSoft GL journal entries. The accounting tables are referenced on the accounting entry definition defined for each type of accounting entry transaction. The Journal Generator uses new accounting entry definitions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

Retail Merchandise Procure to Pay Business Process Flow

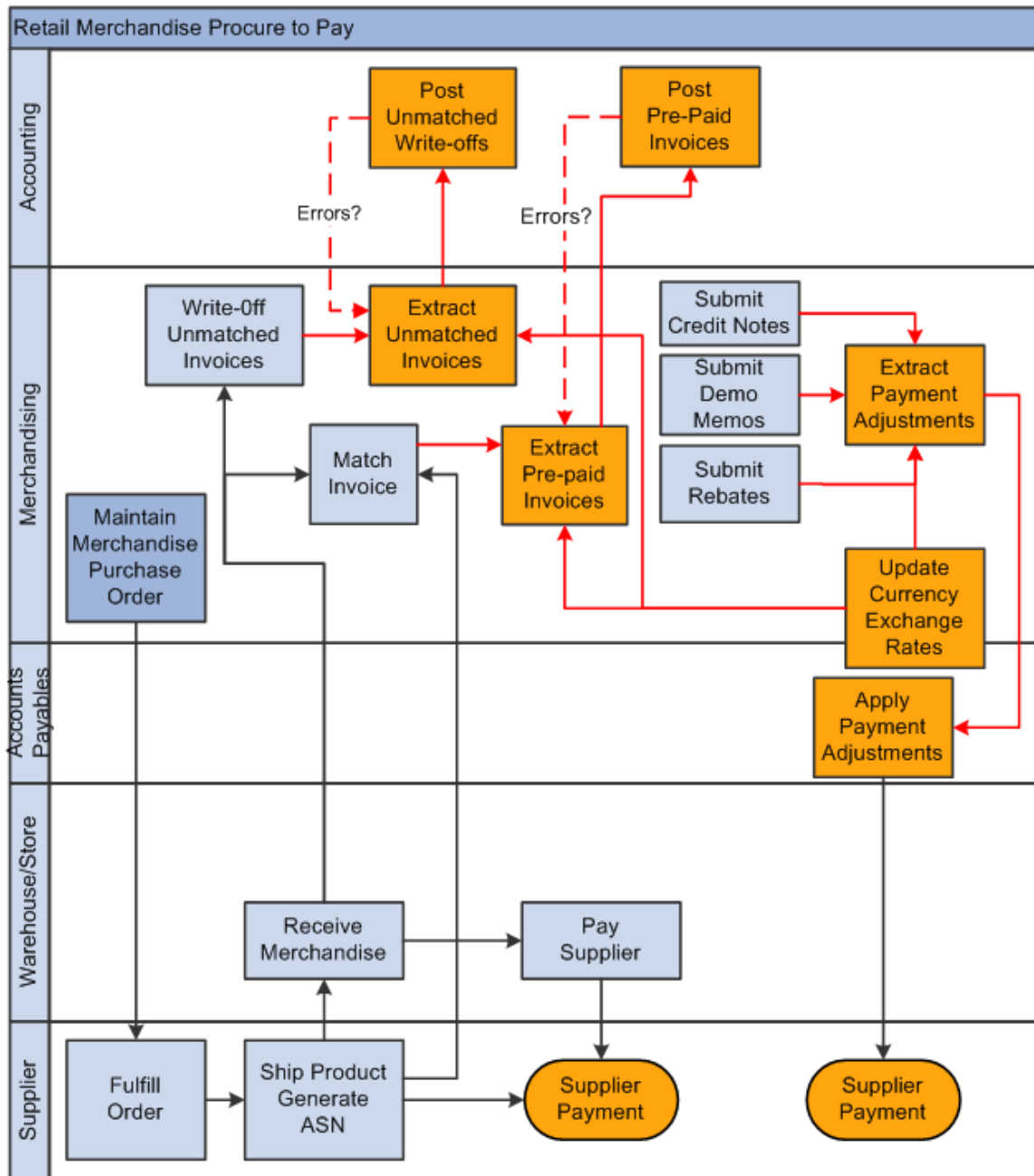
The Retail Merchandise Procure to Pay business process consists of these integration flows:

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

This diagram illustrates the Retail Merchandise Procure to Pay business process flow:



Retail Merchandise Procure to Pay business process flow (1 of 2)



Retail Merchandise Procure to Pay business process flow diagram (2 of 2)

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

Solution Assumptions and Constraints

These are the assumptions and constraints for this PIP:

PeopleSoft GL

1. The PeopleSoft applications are implemented prior to the implementation of 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 do not need 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 do not need to persist the structure in the middle layer. Each transaction message accounts for the setID business unit association.
3. Oracle Retail manually creates and stores the valid segment (ChartField) combinations in the appropriate GL Cross Reference tables (ReSA, RMS, and 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 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.
2. Oracle Retail has no concept of effective dates and therefore only cares about current rates. During implementation, users need to subscribe only to the current published records from PeopleSoft.
3. The currency amounts precision is limited to 3 digits.

Calendar

1. The PeopleSoft applications use multiple calendars. In this PIP, both PeopleSoft and Oracle Retail use a single calendar.
2. When a calendar is established, switching the calendar is not supported in this integration.

Sales Audit/stock ledger Transactions

1. Oracle Retail sends the accounting date and the transaction date with its transactions. These dates should not be changed or manipulated in PeopleSoft.
2. Accounting entry errors that are found from accounting entries are handled manually on both the Oracle Retail and PeopleSoft side.
3. Use or sales tax accounting information is passed as part of the accounting entries between Oracle Retail and PeopleSoft.
4. Value-added tax (VAT) is calculated in Oracle Retail. PeopleSoft Journal Generator does not recalculate VAT. VAT calculation is passed as a part of the accounting entry.
5. Oracle Retail stock ledger determines the valuation of inventory for merchandise being directly procured. This information is passed to PeopleSoft as the accounting entries.

6. RMS, through the Retail stock ledger, provides PeopleSoft with the value of ending inventory at cost using the method that the retailer indicates (cost method or retail method of accounting) by means of an adjusting entry.
7. Accounting entries cannot be loaded twice into PeopleSoft. When the journals are created, a status field called Distribution Status exists. This field changes to D when the Journals are created. So 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 ChartFields (segments) are renamed or changed as to their name and not their value or new ChartFields are created, this is considered a customization of the middle layer.
3. RIB is required for all message-based integration to Oracle Retail.

Constraints

1. Oracle Retail and PeopleSoft applications do not support modifying date effectively by time zones.
2. PeopleSoft Financials can handle only three decimal places in amount fields. For this integration, no currency in Oracle Retail should 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 require reimplementing of the Integration Pack.
4. AIA uses the location ID as a key and when it changes, AIA cannot associate it with the old 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 the respective chapters.

Chapter 2: Life Cycle Management for Reference Data

This chapter discusses these process integrations:

- Currency exchange rate integration
- Payment term integration
- Suppliers information integration

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 (RMS) and discusses:

- Currency exchange rate integration details
- PeopleSoft GL interfaces
- Oracle Retail Integration Bus (RIB) interfaces
- Data requirements
- Core Application Integration Architecture (AIA) components
- Integration services

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 RMS: Enables the loading of all current effective dated currency exchange rates from PeopleSoft GL to RMS for a new instance (logical or physical) of RMS.
- Incremental creation and updates of currency exchange rates from PeopleSoft GL to RMS: Enables the synchronization of incremental creation and updates of the current effective dated currency exchange rates from PeopleSoft GL to RMS.

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

Prerequisites

These are the prerequisites for the process integration for currency exchange rates.

1. The list of valid currencies is maintained separately in each application and synchronized manually.
2. Currency exchange rate types are mapped 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 by means of a local process.

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 AIA layer. Together they need to come up with the COMMON value.

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.

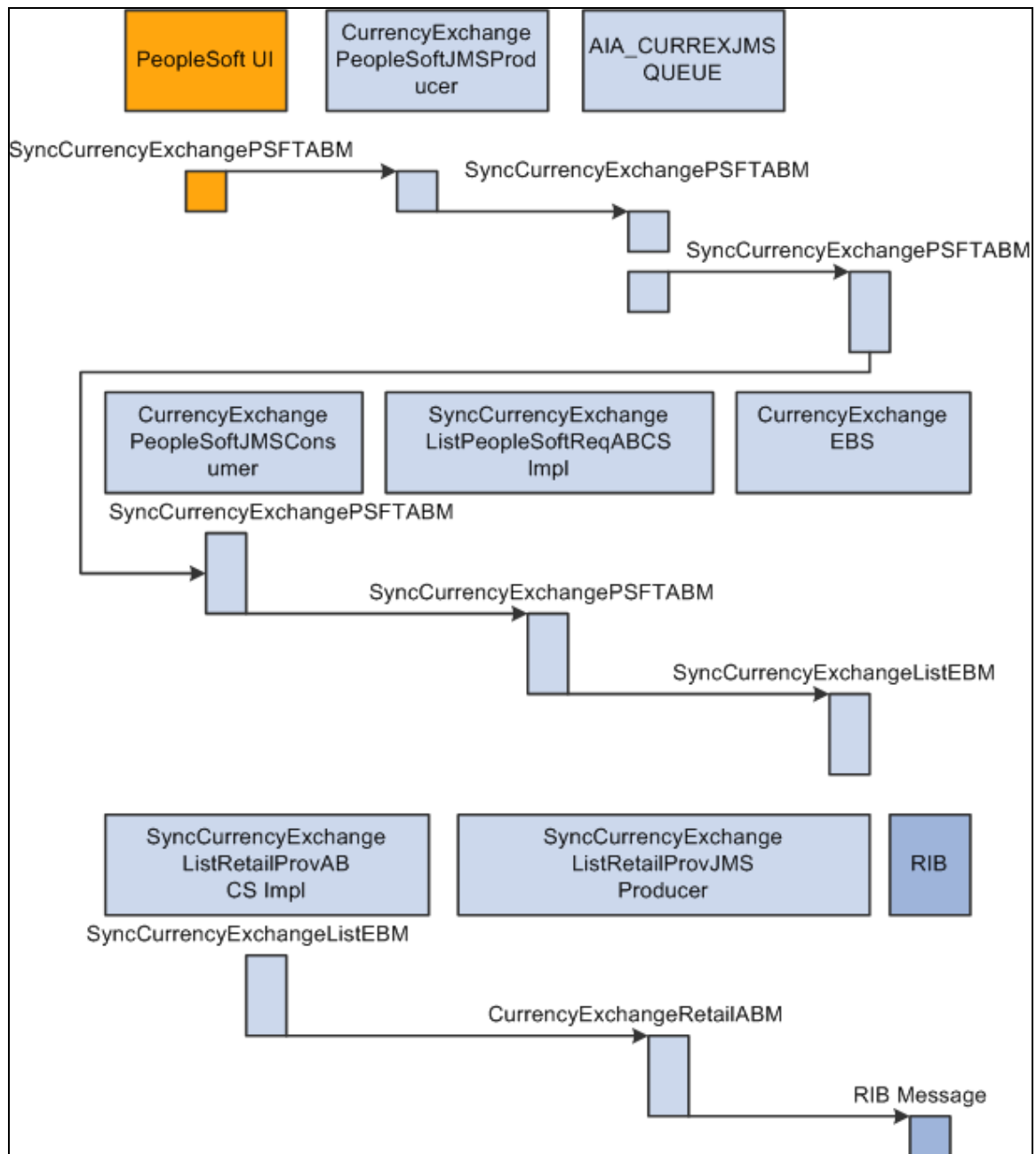
5. When the PeopleSoft currency exchange rates are published, the current effective dated currency exchange rates are synchronized with the Oracle Retail exchange rates.
6. Oracle Retail accepts all records from PeopleSoft except the records with the previous dates (prior to their vdate).
7. The PeopleSoft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

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

9. Retail RIB Error Hospital holds all the Oracle Retail side errors and handles any notification on their side.

10. If a value is not present for Oracle Retail in the CURRENCYEXCHANGE_CONVERSIONTYPECODE DVM, the AIA layer filters the record and does not send it to Oracle Retail. A message is sent to let users know that the value used for this DVM is not in the DVM. But, no error is generated for this.
11. If the value is missing for the CURRENCY_CODE DVM, the 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 its base currency and ignores other exchange rates.
Oracle Retail handles this internally. The AIA layer does not filter exchange rates.

This diagram illustrates the currency exchange rate integration flow:



Currency exchange rate integration flow

Currency Exchange Rate Integration Details

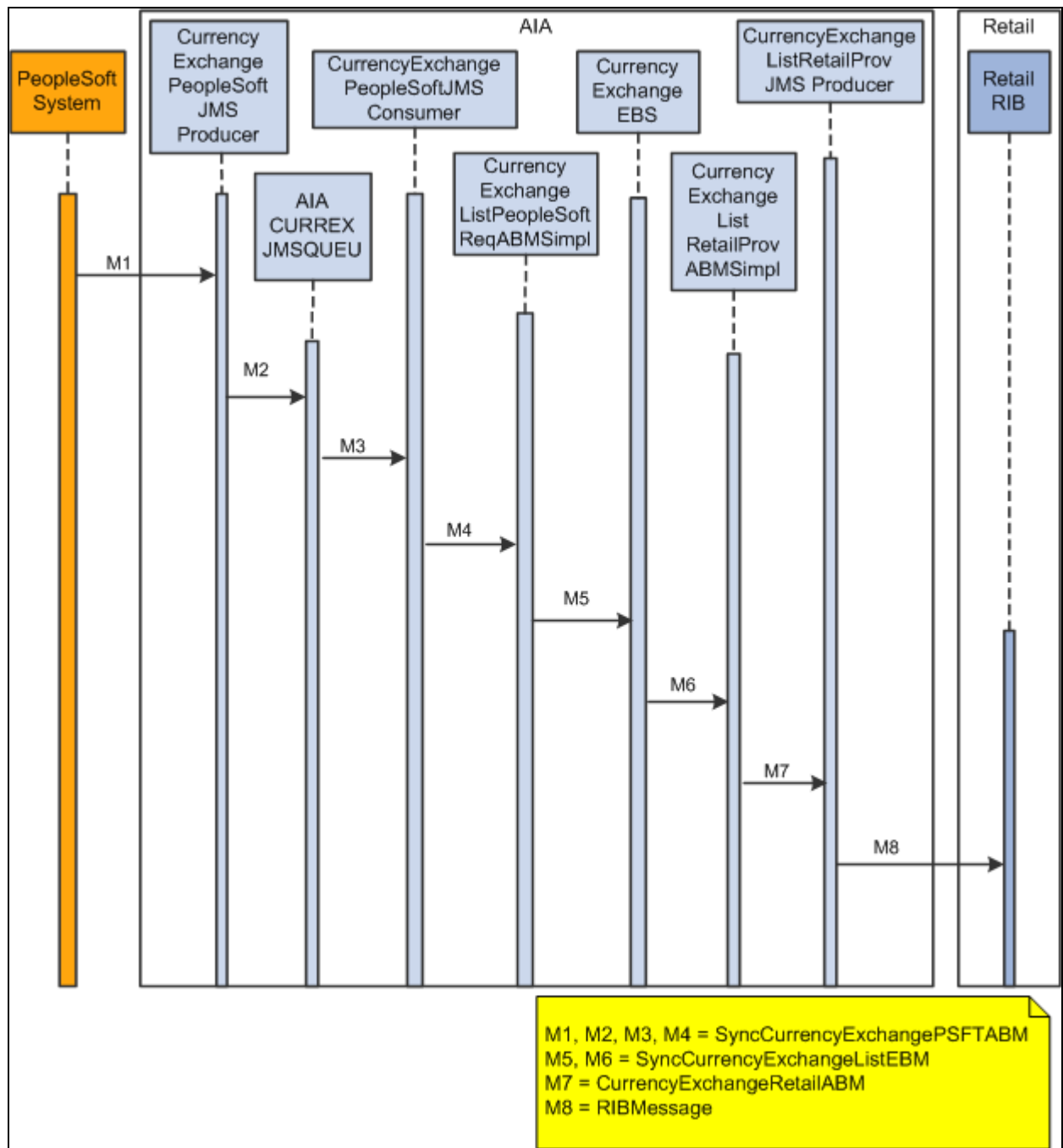
The integration flow uses these services:

- CurrencyExchangePeopleSoftJMSProducer
- CurrencyExchangePeopleSoftJMSConsumer
- SyncCurrencyExchangeListPeopleSoftReqABCSImpl

- CurrencyExchangeEBS
- SyncCurrencyExchangeListRetailProvABCSImpl
- SyncCurrencyExchangeListRetailProvJMSProducer

Sequence Diagram

This sequence diagram illustrates the currency exchange rate integration flow:



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 invokes the CurrencyExchangePeopleSoftJMSProducer service with SyncCurrencyExchangeListPSFTABM whenever a currency exchange rate is created or updated.

This service populates JMSCorrelationID and puts the message in AIA_CURREXJMSQUEUE.
2. The CurrencyExchangePeopleSoftJMSConsumer service picks up the message from AIA_CURREXJMSQUEUE based on JMSCorrelationID 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 an EBM. PeopleSoft ABM does not have some details such as the EBM header and the 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.

PeopleSoft GL Interfaces

Outbound Interactions

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

For more information, see the *Enterprise PeopleTools PeopleBook 9.0*, “PeopleSoft Integration Broker,” Managing Service Operations

Oracle RIB Interfaces

Inbound Interactions

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

For more information, see the RIB Integration Guide

Data Requirements

No data requirements exist for this process integration.

Core AIA Components

The currency exchange rate integration uses these components:

- CurrencyExchangeEBO
- CurrencyExchangeEBM
- CurrencyExchangeEBS

The core EBO and EBM XSD files can be located by EBO within this parent folder:
http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core EBS WSDL files can be located by EBO within this parent folder:
http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation of individual EBOs, click the EBO Name link on the Integration Scenario Summary page in the Oracle AIA Console. You can also use the Integration Scenario Summary page to search for and view integration scenarios that use a particular EBO or Enterprise Business Service (EBS).

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” “Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

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

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Extensibility for Oracle AIA Artifacts,” Extending EBOs

Integration Services

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

- CurrencyExchangePeopleSoftJMSProducer
- CurrencyExchangePeopleSoftJMSConsumer
- SyncCurrencyExchangeListPeopleSoftReqABCSImpl
- CurrencyExchangeEBS
- SyncCurrencyExchangeListRetailProvABCSImpl
- SyncCurrencyExchangeListRetailProvJMSProducer

You can use the Integration Scenario Summary page in the Oracle AIA Console to search for and view integration scenarios that use a particular ABC service.

For more information, see Oracle Application Integration Architecture – Foundation Pack: Core Infrastructure Components Guide, “Using the BSR,” Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

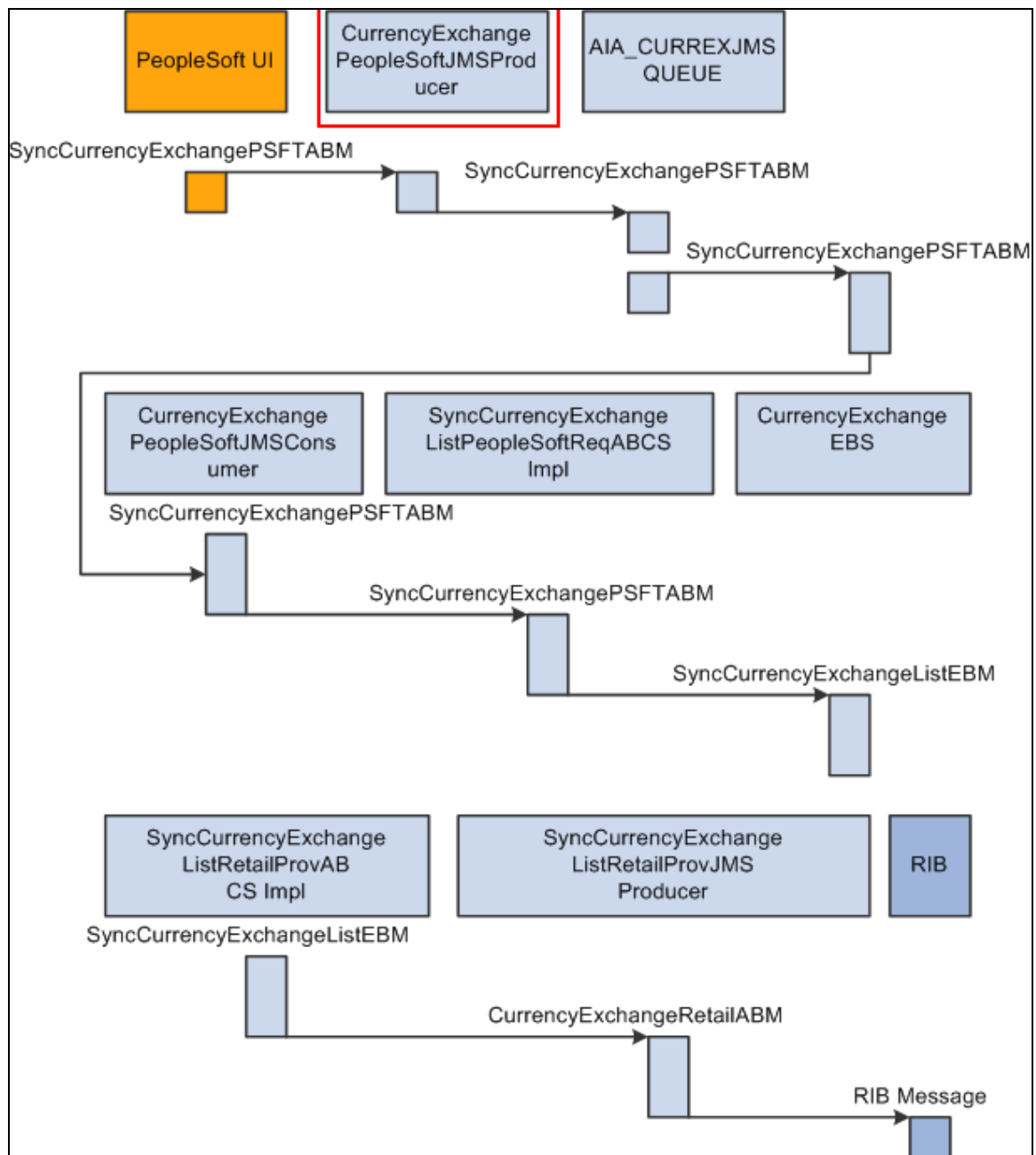
CurrencyExchangePeopleSoftJMSProducer

The CurrencyExchangePeopleSoftJMSProducer service is a Business Process Execution Language (BPEL) process. The PeopleSoft system invokes this service when:

- A new currency exchange rate is created.
- An existing currency exchange rate is updated.
- A synchronization operation exists for a bulk load of currency exchange rates with SyncCurrencyExchangeListPSFTABM.

This service populates the JMSCorrelationID and invokes SyncCurrencyExchangeListPeopleSoftJMSProducer, which then puts the message in the AIA_CURREXJMSQUEUE.

This diagram illustrates the relationship of the CurrencyExchangePeopleSoftJMSProducer with the other services in the integration flow:

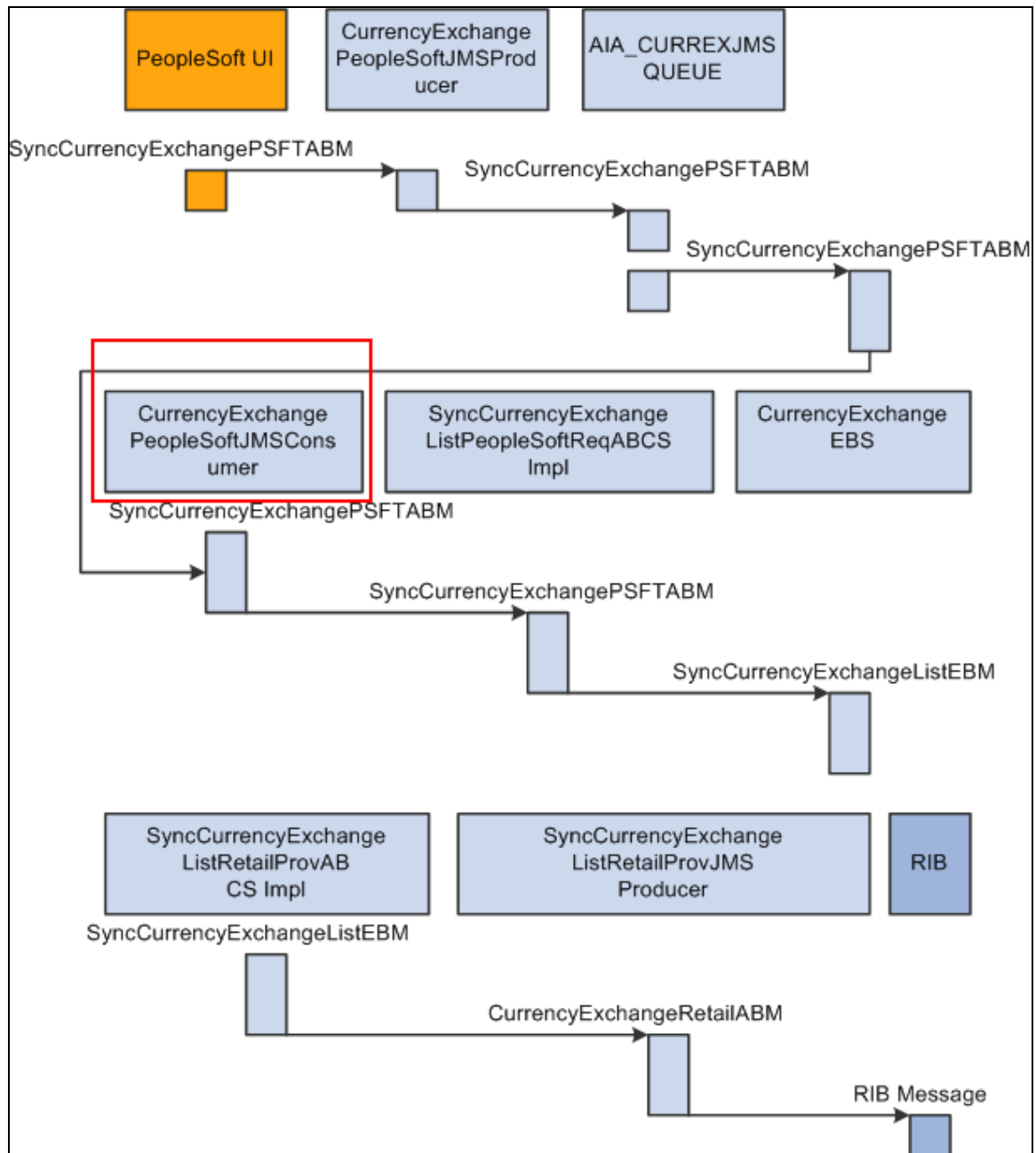


CurrencyExchangePeopleSoftJMSProducer

CurrencyExchangePeopleSoftJMSConsumer

CurrencyExchangePeopleSoftJMSConsumer is an Enterprise Service Bus (ESB) service. It has a JMS adapter called SyncCurrencyExchangeListPeopleSoftJMSConsumer. This adapter listens to the AIA_CURREXJMSQUEUE and picks up the messages for which JMSCorrelationID is SYNC. This invokes SyncCurrencyExchangeListPeopleSoftReqABCImpl with the SyncCurrencyExchangeListPSFTABM.

This diagram illustrates the relationship of the CurrencyExchangePeopleSoftJMSConsumer with the other services in the integration flow:

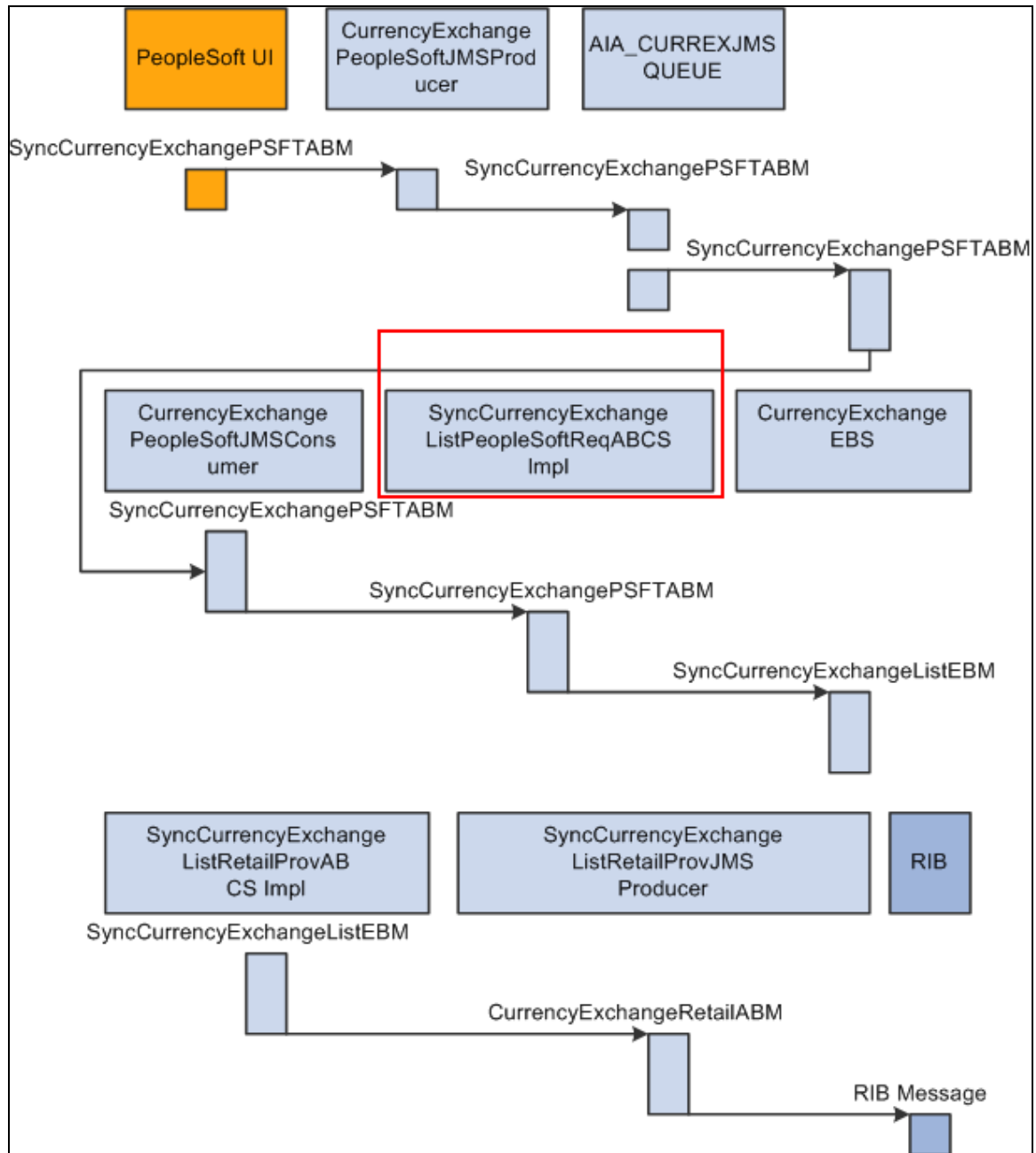


CurrencyExchangePeopleSoftJMSConsumer

SyncCurrencyExchangeListPeopleSoftReqABCImpl

The SyncCurrencyExchangeListPeopleSoftReqABCImpl service is a BPEL process. This is a thin requester, which receives SyncCurrencyExchangeListPSFTABM from SyncCurrencyExchangePeopleSoftConsumer. This message is transformed to SyncCurrencyExchangeListEBM, which populates Enterprise Business Message (EBM) Header and populates the XREF table. No DVM lookups are required for this message because the PeopleSoft system sends all the AIA Common values. This SyncCurrencyExchangeListEBM is sent to CurrencyExchangeEBS.

This diagram illustrates the relationship of the SyncCurrencyExchangeListPeopleSoftReqABCSImpl with the other services in the integration flow:

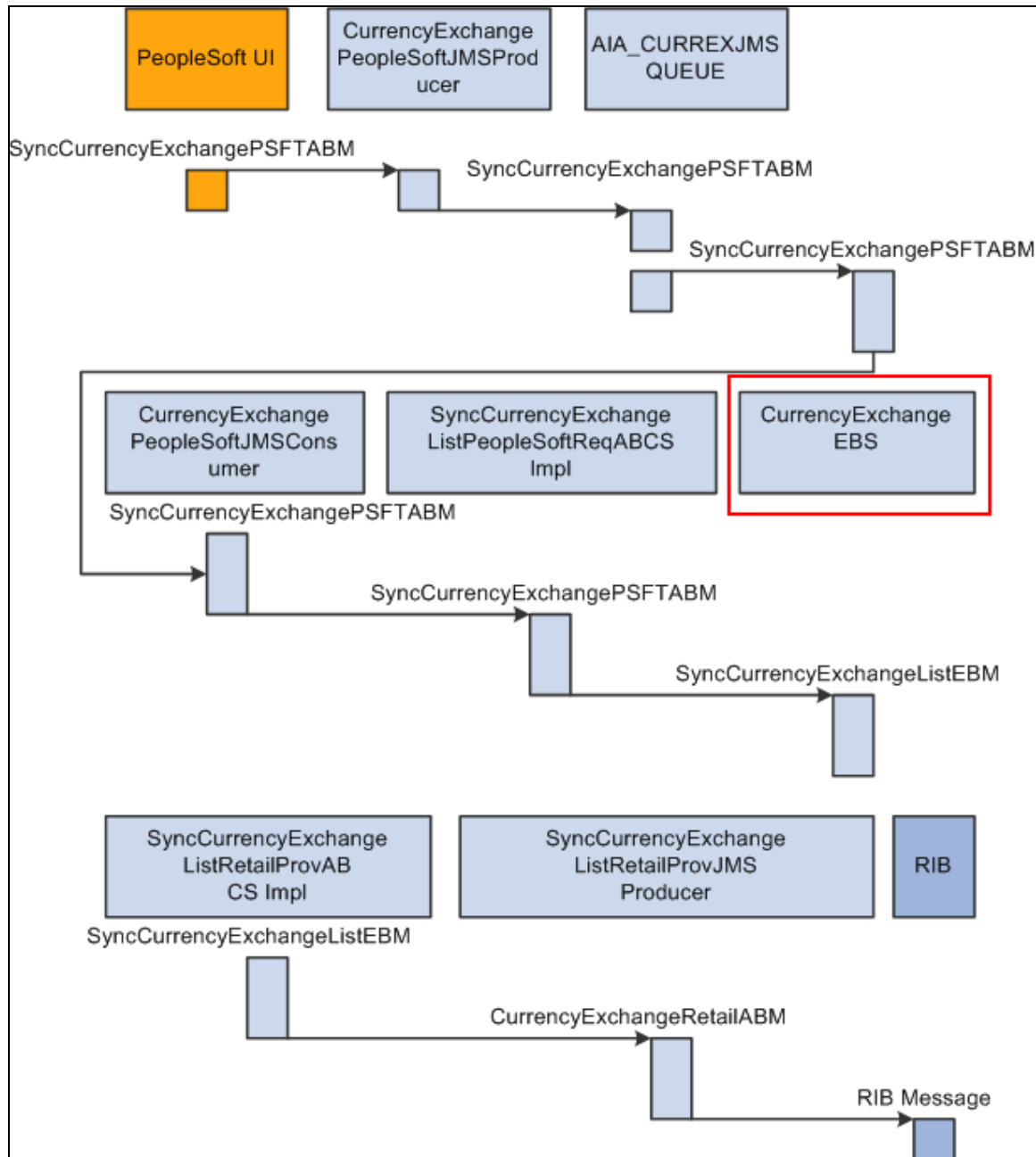


[SyncCurrencyExchangeListPeopleSoftReqABCSImpl](#)

CurrencyExchangeEBS

CurrencyExchangeEBS routes all currency exchange related actions, such as create and update of currency exchange rate as Sync Currency Exchange Rate message to SyncCurrencyExchangeListRetailProvABCSImpl or Composite Application Validation System (CAVS) based on the filter condition and operations. For this particular integration, only 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 create or update action.

This diagram illustrates the relationship of the CurrencyExchangeEBS with the other services in the integration flow:



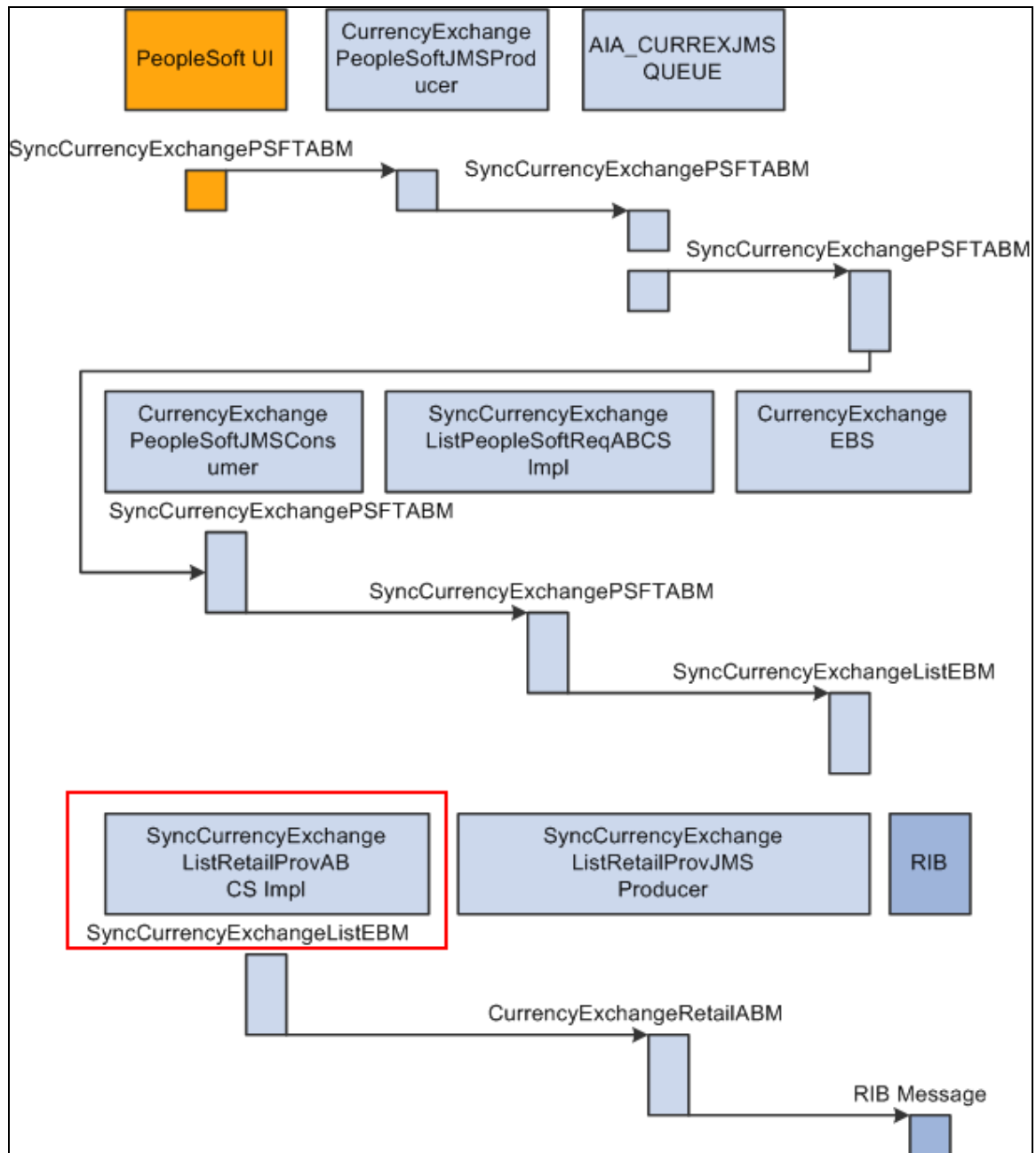
CurrencyExchangeEBS

For more information about this EBS, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Designing and Developing EBSs” and *Oracle Application Integration Architecture – Foundation Pack: Concepts and Technologies Guide*, “Understanding EBSs”

SyncCurrencyExchangeListRetailProvABCImpl

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

This diagram illustrates the relationship of the SyncCurrencyExchangeListRetailProvABCSImpl with the other services in the integration flow:



[SyncCurrencyExchangeListRetailProvABCSImpl](#)

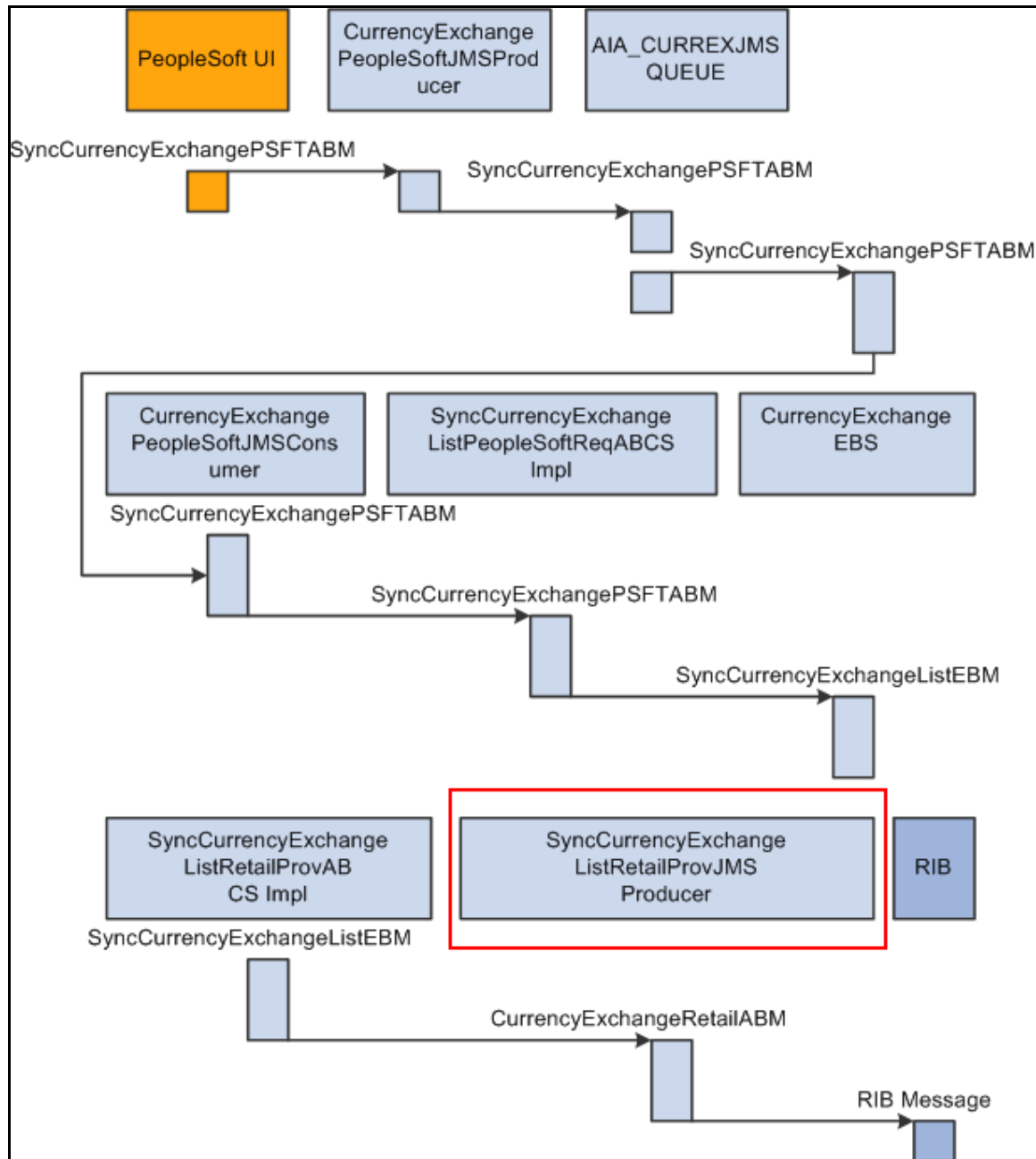
SyncCurrencyExchangeListRetailProvJMSProducer

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

This service performs these actions:

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

This diagram illustrates the relationship of the SyncCurrencyExchangeListRetailProvJMSProducer with the other services in the integration flow:



[SyncCurrencyExchangeListRetailProvJMSProducer](#)

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 Retail Merchandising System (RMS) and discusses:

- Payment term integration details
- PeopleSoft Payables interfaces
- RMS interfaces
- Data requirements
- Core AIA components
- Integration services

Payment Term Integration Overview

In the integrated environment, PeopleSoft Enterprise Financials acts as a payable and accounting engine with RMS for supplier payment, merchandise write-offs, and prepaid adjustments. It eliminates the need for manual reentry of the reference data from PeopleSoft Payables to the RMS. This benefits retailers by 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. 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 RMS through these integration flows:

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

For more information about payment terms, see *PeopleSoft Enterprise Source to Settle Common Information 9.0 PeopleBook*, “Defining Procurement Options,” Defining Payment Terms

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

Prerequisites

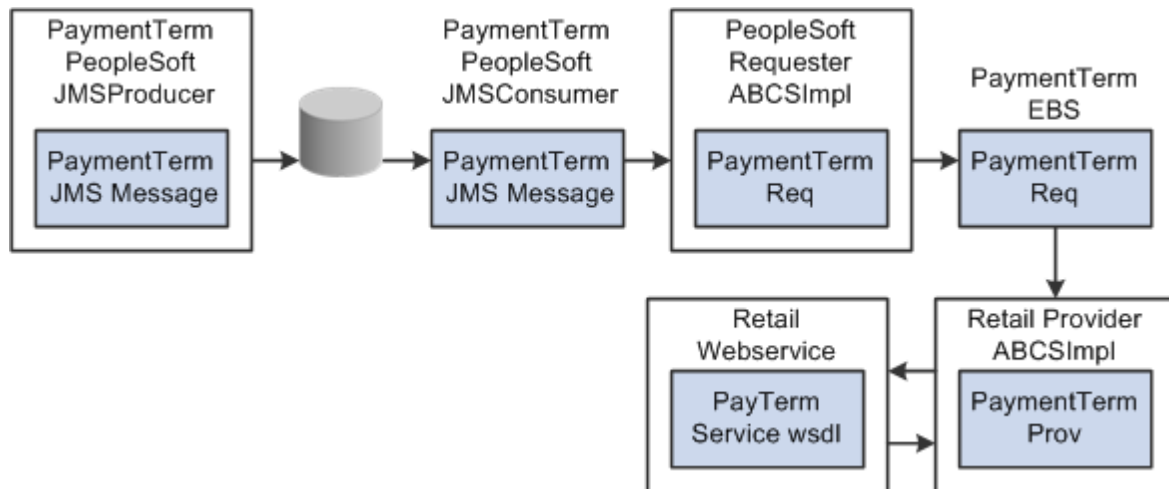
No prerequisites exist for the process integration for payment term.

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, it is rejected by Oracle Retail.
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. It is recommended that the same payment terms code value not be set up under multiple setID 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 prior to the Sync operation being performed.
The last update to a payment term in 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 RMS.
9. Payment term integration occurs prior to supplier initial load and manual setup of freight term in RMS.
10. The CreatePaymentTermRetailProvABCSImpl, SyncPaymentTermListRetailProvABCSImpl, and UpdatePaymentTermRetailProvABCSImpl assign the "Creation Date Time", originating from the PeopleSoft ABCS implementation, to the end_date_Active if the status code from PeopleSoft is an "I".

This diagram illustrates the payment term integration flow:



Payment term integration flow

Payment Term Integration Details

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

- PaymentTermPeopleSoftJMSProducer
- PaymentTermPeopleSoftJMSConsumer
- PaymentTermEBS
- PayTermService

These services are specific to Sync payment term integration flow:

- SyncPaymentTermListPeopleSoftReqABCSImpl
- SyncPaymentTermListRetailProvABCSImpl

These services are specific to create payment term integration flow:

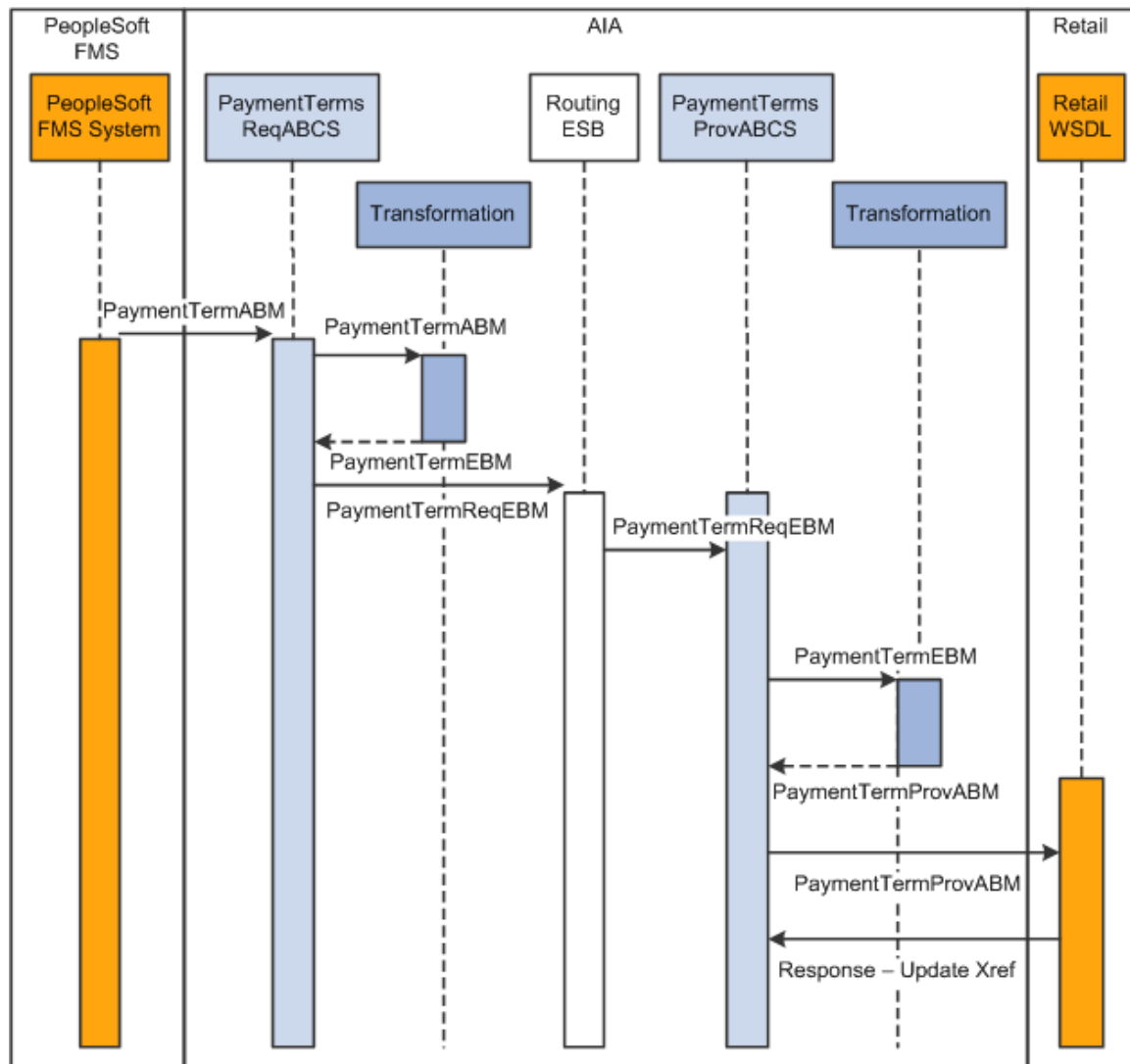
- CreatePaymentTermPeopleSoftReqABCSImpl
- CreatePaymentTermRetailProvABCSImpl

These services are specific to update payment term integration flow.

- UpdatePaymentTermPeopleSoftReqABCSImpl
- UpdatePaymentTermRetailProvABCSImpl

Sequence Diagram

This diagram illustrates the synchronize payment term sequence flow:



Synchronize payment term sequence diagram

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

When you initiate the process:

1. PeopleSoft invokes the `PaymentTermPeopleSoftJMSProducer` service with `SyncPaymentTermListPSFTABM`, `CreatePaymentTermPSFTABM`, and `UpdatePaymentTermPSFTABM`.

This service puts this ABM in the `AIA_PAYMENTTERMJMSQUEUE` with `JMSCorrelationID` set to `Sync/Create/Update`.
2. The `PaymentTermPeopleSoftJMSConsumer` listens on the `AIA_PaymentTermJMSQueue` and invokes `SyncPaymentTermListPeopleSoftReqABCImpl` in case of `SyncPaymentTermListPSFTABM`, `CreatePaymentTermPeopleSoftReqABCImpl` in case of `CreatePaymentTermPSFTABM`, and `UpdatePaymentTermPeopleSoftReqABCImpl` in case of `UpdatePaymentTermPSFTABM`.

3. For the initial load, PeopleSoft invokes the SyncPaymentTermListPeopleSoftReqABCSEImpl 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 a new payment term is created in PeopleSoft, CreatePaymentTermPeopleSoftReqABCSEImpl is invoked 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 a payment term in PeopleSoft is updated, the UpdatePaymentTermPeopleSoftReqABCSEImpl service is invoked 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 SyncPaymentTermListRetailProvABCSEImpl, CreatePaymentTermRetailProvABCSEImpl, UpdatePaymentTermRetailProvABCSEImpl, or CAVS based upon the operation and filter condition.

7. The SyncPaymentTermRetailProvABCSEImpl service transforms SyncPaymentTermListEBM to Retail ABM.

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

8. The CreatePaymentTermRetailProvABCSEImpl service transforms CreatePaymentTermEBM to Retail ABM.

Transformation synchronously invokes the create PayTermService web service from RMS.

9. The UpdatePaymentTermRetailProvABCSEImpl transforms UpdatePaymentTermEBM to Retail ABM.

Transformation synchronously invokes the update PayTermService web service from RMS.

Note: The structure of the PeopleSoft ABM is very close to an EBM. PeopleSoft ABM does not have some details such as the EBM header and the 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.

Multiple Language Support

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

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

Because Oracle Retail supports only one primary language per instance, the 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 look up based on the common EBM Header. If the DVM language code look up 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, RMS can support only single-tier payment terms. The AIA integration for payment term passes only translatable fields from PeopleSoft to Oracle Retail for the primary language of the Oracle Retail instance being integrated. This applies to the incremental creates and updates of payment terms.

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

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.

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

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, Volume 2, "Web Services Overview"

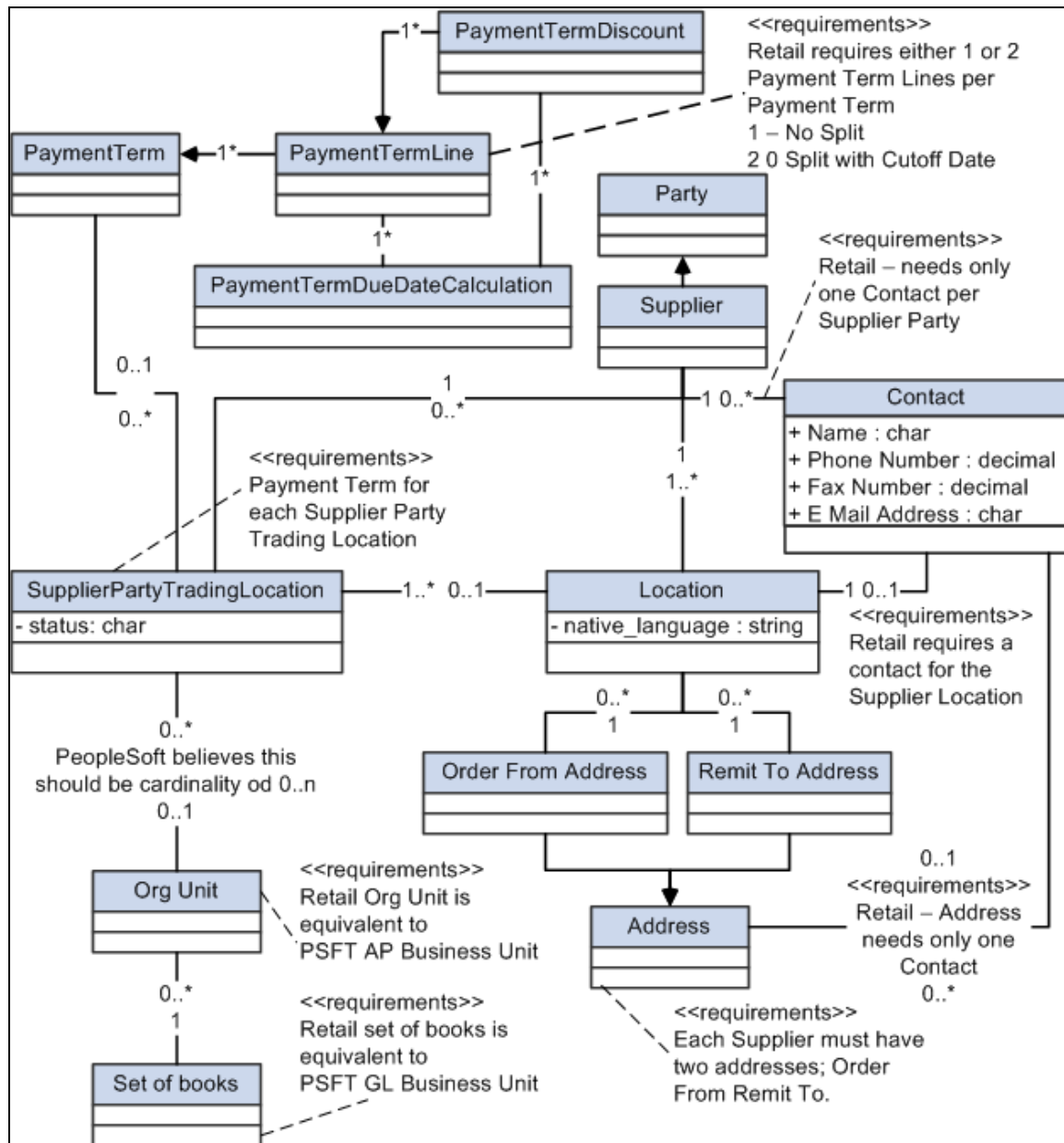
Data Requirements

The process integration for payment term requires these data:

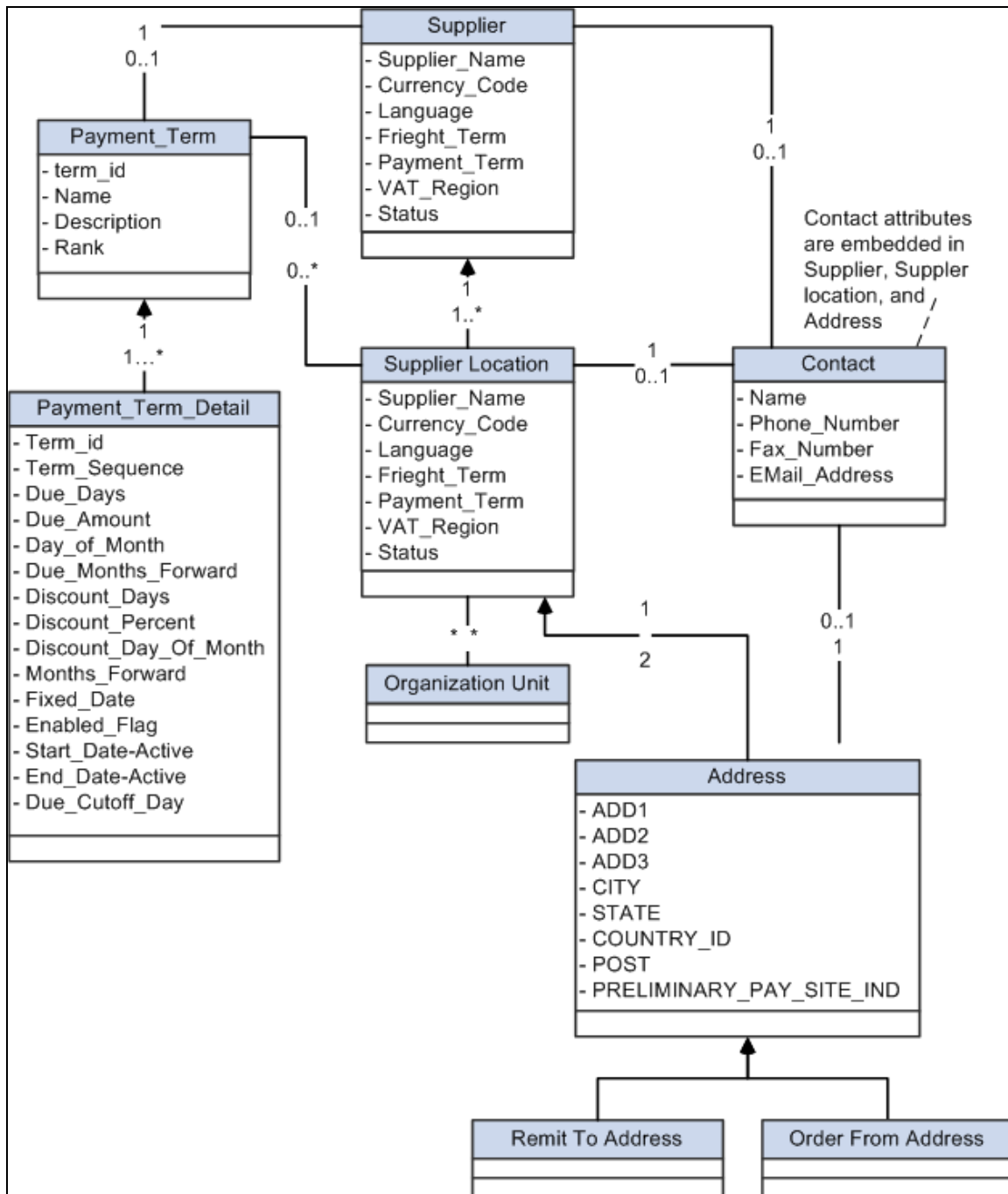
- In PeopleSoft Payables, payment terms must be assigned to at least one set of books (GL business unit or AP business unit or both) upon creation and generation of ABM.
- Assignment of payment terms to additional set of books (GL business unit or AP business unit or both) generates an ABM for this event.

Class Diagram

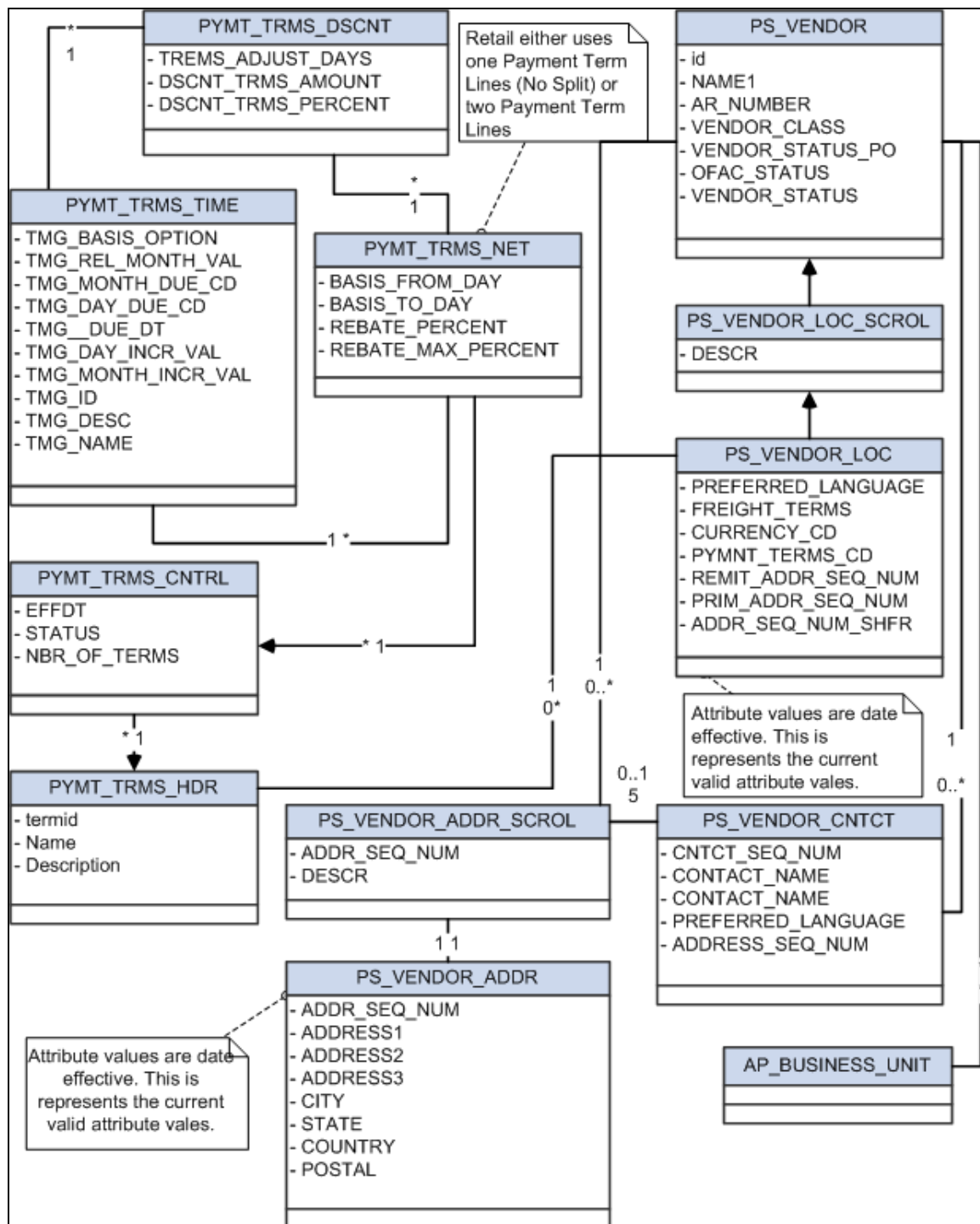
These are class diagrams for payment term integration:



Class diagram 1 of 3



Class diagram 2 of 3



Class diagram 3 of 3

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:
[http://\[HOST:PORT\]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/](http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/).

The core EBS WSDL files can be located by EBO within this parent folder:
[http://\[HOST:PORT\]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/](http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/).

For detailed documentation of individual EBOs, click the EBO Name link on the Integration Scenario Summary page in the Oracle AIA Console. You can also use the Integration Scenario Summary page to search for and view integration scenarios that use a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

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

For more information, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Extensibility for AIA Artifacts,” Extending EBOs

Integration Services

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

- PaymentTermPeopleSoftJMSProducer
- PaymentTermPeopleSoftJMSConsumer
- SyncPaymentTermListPeopleSoftReqABCSImpl
- CreatePaymentTermPeopleSoftReqABCSImpl
- UpdatePaymentTermPeopleSoftReqABCSImpl
- PaymentTermEBS
- SyncPaymentTermListRetailProvABCSImpl
- CreatePaymentTermRetailProvABCSImpl
- UpdatePaymentTermRetailProvABCSImpl
- PayTermService

You can use the Integration Scenario Summary page in the Oracle AIA Console to search for and view integration scenarios that use a particular ABC service.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

PaymentTermPeopleSoftJMSProducer

PeopleSoft invokes this service when a new payment term is created or an existing payment term is updated, or for synchronization of a bulk load of payment terms. This process receives CreatePaymentTermPSFTABM, UpdatePaymentTermPSFTABM, and SyncPaymentTermListPSFTABM from PeopleSoft for create, update, and synch operations respectively.

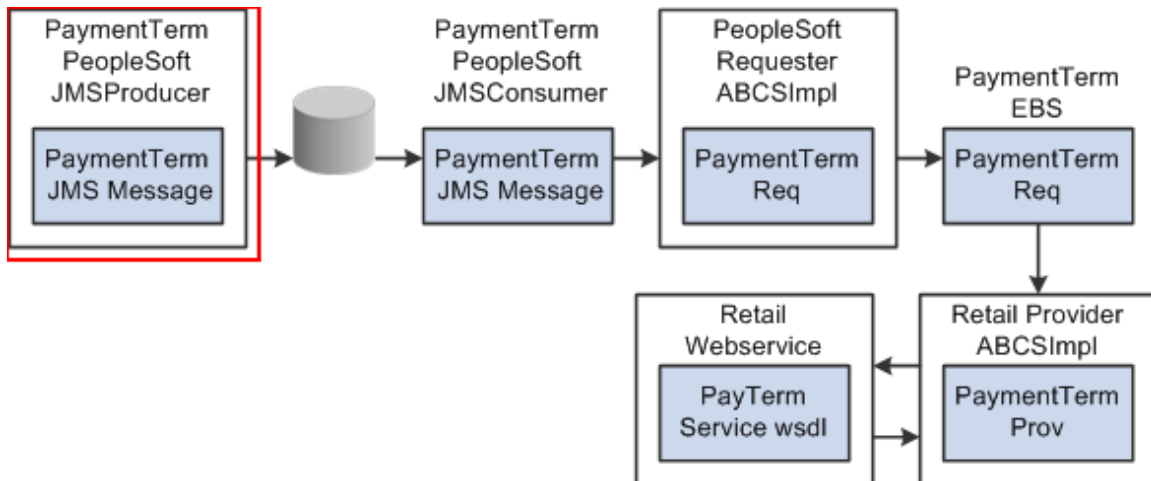
This service is a BPEL process with SyncPaymentTermList, CreatePaymentTerm, and UpdatePaymentTerm operations, which route the SyncPaymentTermListPSFTABM, CreatePaymentTermPSFTABM, and UpdatePaymentTermPSFTABM to the corresponding JMS Adapter after populating the JMSCorrelationID.

This service invokes These JMS Adapters:

- CreatePaymentTermPeopleSoftJMSProducer
- UpdatePaymentTermPeopleSoftJMSProducer
- SyncPaymentTermListPeopleSoftJMSProducer

These adapters put the respective messages in AIA_PaymentTermJMSQueue.

This diagram illustrates the relationship of the PaymentTermPeopleSoftJMSProducer with the other services in the integration flow:

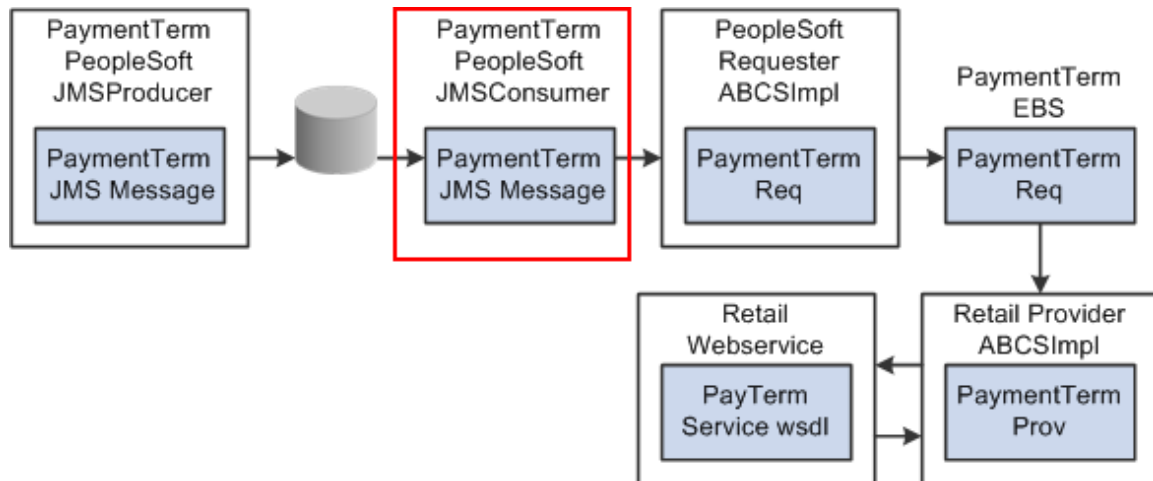


PaymentTermPeopleSoftJMSProducer

PaymentTermPeopleSoftJMSConsumer

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

This diagram illustrates the relationship of the PaymentTermPeopleSoftJMSConsumer with the other services in the integration flow:



PaymentTermPeopleSoftJMSConsumer

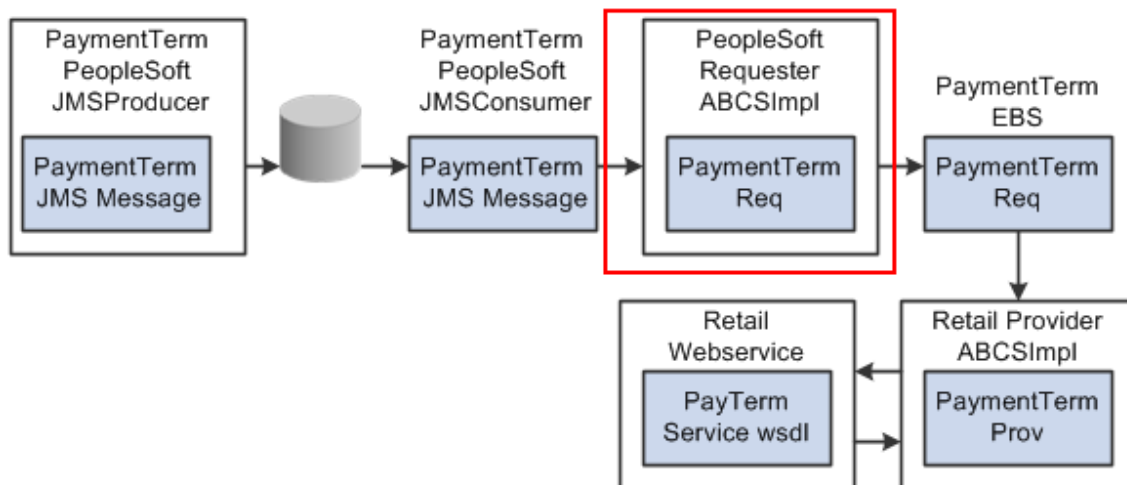
SyncPaymentTermListPeopleSoftReqABCSImpl

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 the EBM Header population, XREF population, and setting the environment code to CAVS or PRODUCTION based on RouteToCAVS property in AIAConfigurationProperties file. The Request message adheres to 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 accepts SyncPaymentTermListEBM as input and returns the same.

This diagram illustrates the relationship of the SyncPaymentTermListPeopleSoftReqABCImpl with the other services in the integration flow:



[SyncPaymentTermListPeopleSoftReqABCImpl](#)

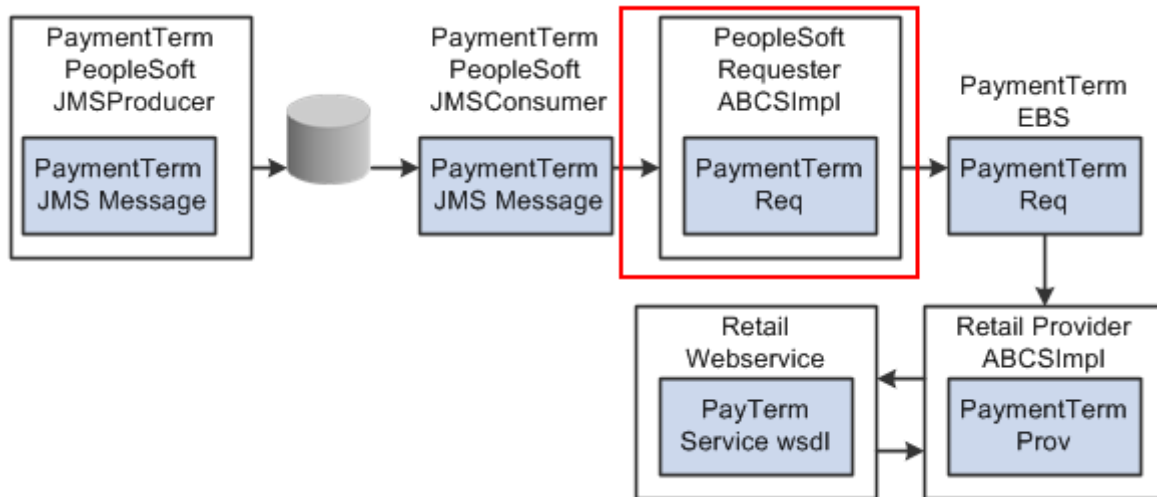
CreatePaymentTermPeopleSoftReqABCImpl

`CreatePaymentTermPeopleSoftReqABCImpl` is a thin requester that receives `CreatePaymentTermPSFTABM` from PeopleSoft. This input is transformed to `CreatePaymentTermEBM`. It populates EBM Header XREF 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 accepts `CreatePaymentTermEBM` as input and returns the same.

This diagram illustrates the relationship of the CreatePaymentTermPeopleSoftReqABCSImpl with the other services in the integration flow:



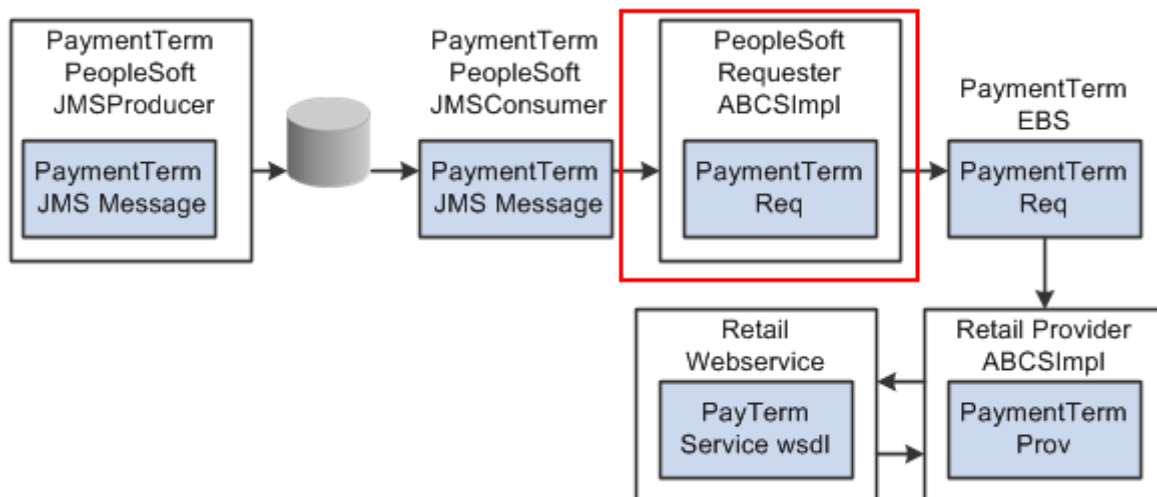
CreatePaymentTermPeopleSoftReqABCSImpl

UpdatePaymentTermPeopleSoftReqABCSImpl

UpdatePaymentTermPeopleSoftReqABCSImpl 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 UpdatePaymentTermPeopleSoftReqABCSImplExt is an extensibility service that enables customers to filter, validate, or augment the input. This accepts UpdatePaymentTermEBM as input and returns the same.

This diagram illustrates the relationship of the UpdatePaymentTermPeopleSoftReqABCSImpl with the other services in the integration flow:

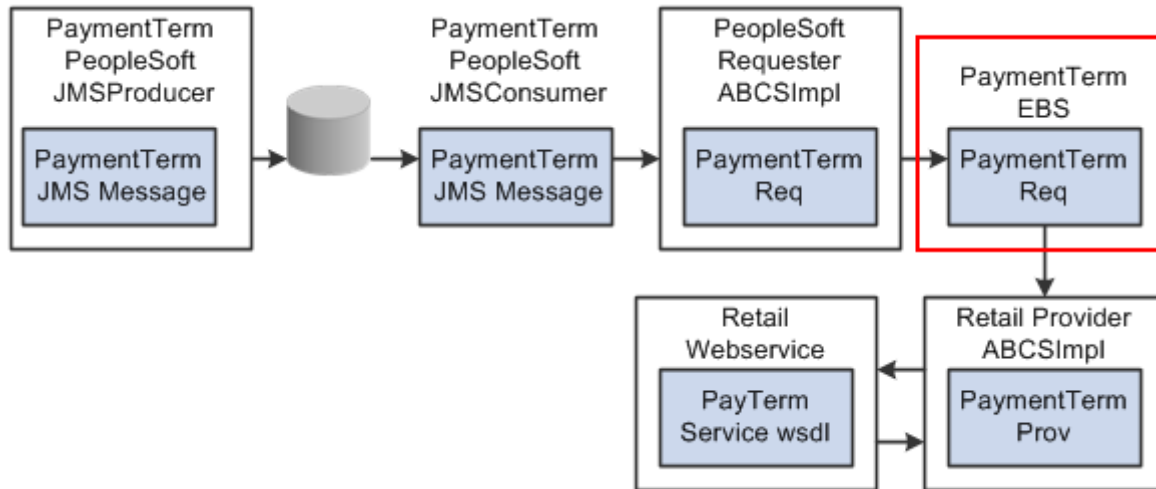


UpdatePaymentTermPeopleSoftReqABCSImpl

PaymentTermEBS

PaymentTermEBS is an Enterprise Business Service. It routes to SyncPaymentTermListRetailProvABCImpl, CreatePaymentTermRetailProvABCImpl, UpdatePaymentTermRetailProvABCImpl, or CAVS based on the operation and filter condition.

This diagram illustrates the relationship of the PaymentTermEBS with the other services in the integration flow:



PaymentTermEBS

For more information about this EBS, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Designing and Developing EBSs” and *Oracle Application Integration Architecture – Foundation Pack: Concepts and Technologies Guide*, “Understanding EBSs”

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.

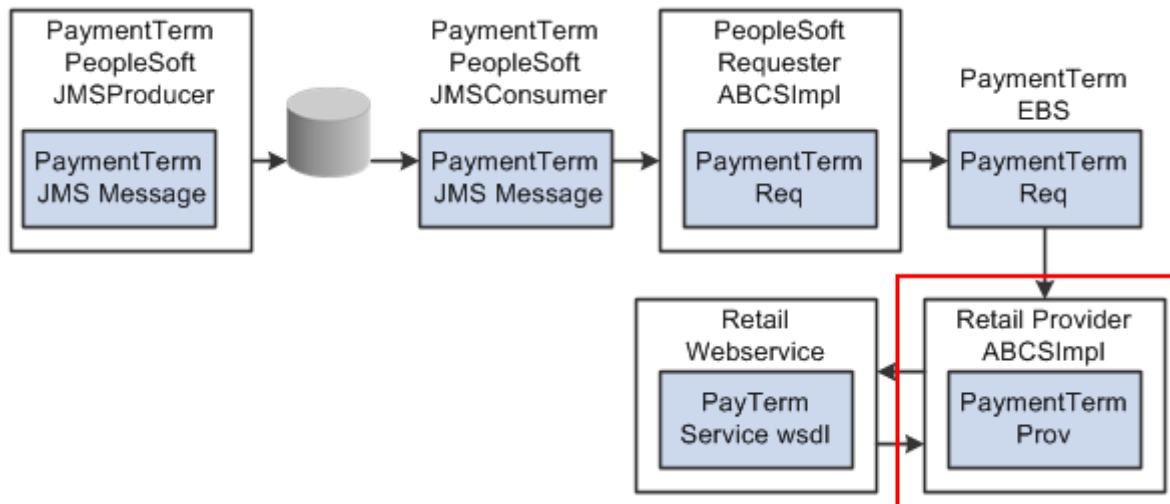
This SyncPaymentTermEBM undergoes one of the four transformations that transform the input to four types of messages that Oracle Retail accepts, based on the header and detail (line) action code. While transforming the data, the DVM looks up the common EBM language and common status codes to determine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

This table illustrates the operations used by this service and related messages:

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 should be used to populate the RETL column in the XREF table PAYMENTTERMLINE_ID.

This diagram illustrates the relationship of the SyncPaymentTermListRetailProvABCSImpl with the other services in the integration flow:



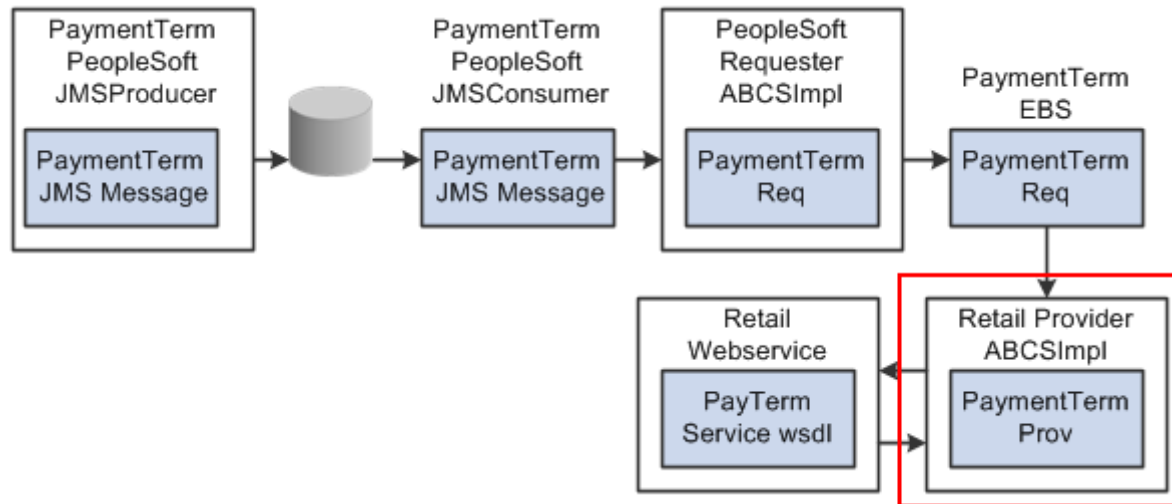
[SyncPaymentTermListRetailProvABCSImpl](#)

CreatePaymentTermRetailProvABCSImpl

CreatePaymentTermRetailProvABCSImpl is a BPEL process. It accepts CreatePaymentTermEBM as input from PaymentTermEBS. This PaymentTermABM is then sent to the PreProcessABM operation of CreatePaymentTermRetailProvABCSImplExt if a service property is set to true in the AIAConfigurationProperties file. This extension service returns the same CreatePaymentTermEBM after processing it. This CreatePaymentTermEBM is transformed 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 should be used to populate the RETL column in the XREF table PAYMENTTERMLINE_ID. While transforming the data, the DVM looks up the common status code to determine the RMS enabled flag value.

This diagram illustrates the relationship of the CreatePaymentTermRetailProvABCImpl with the other services in the integration flow:



CreatePaymentTermRetailProvABCImpl

UpdatePaymentTermRetailProvABCImpl

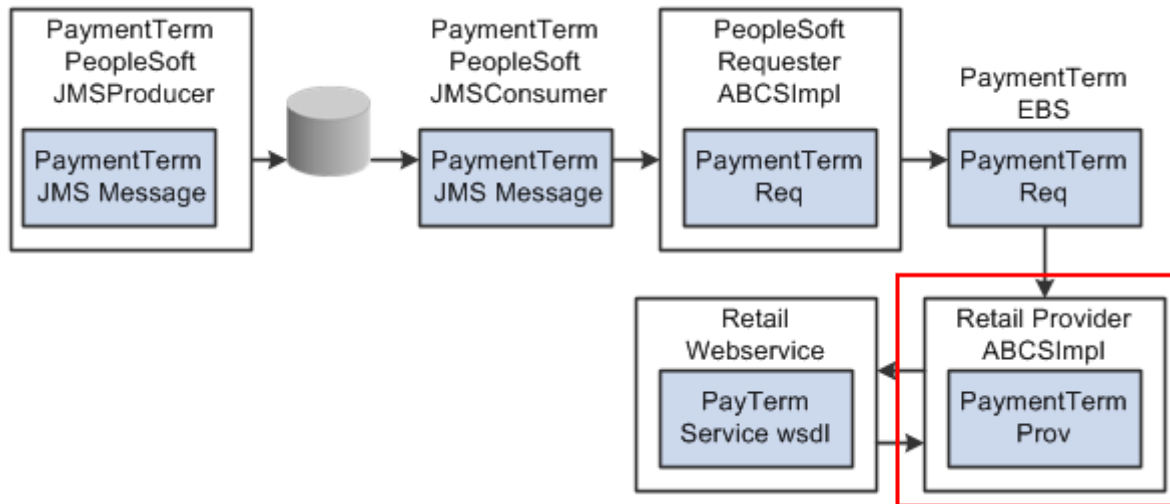
UpdatePaymentTermRetailProvABCImpl is a BPEL process. It accepts UpdatePaymentTermEBM as input from PaymentTermEBS. The PreProcessEBM operation of UpdatePaymentTermRetailProvABCImplExt is invoked with this EBM if a service property in AIAConfigurationProperties file is set to true. This extension service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

This UpdatePaymentTermEBM undergoes one of the three transformations that transform the input to PayTermMod (updateHeaderPayTermDesc), PayTermDtIMod (updatePayTermDesc), or PayTermDtICre (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 RouteToCAVS property is set to false.

The response message from Retail PayTermService, for operation corresponding to PayTermDtICre message, has the "terms-seq", which should be used to populate the RETL column in the XREF table PAYMENTTERMLINE_ID.

This diagram illustrates the relationship of the UpdatePaymentTermRetailProvABCSImpl with the other services in the integration flow:

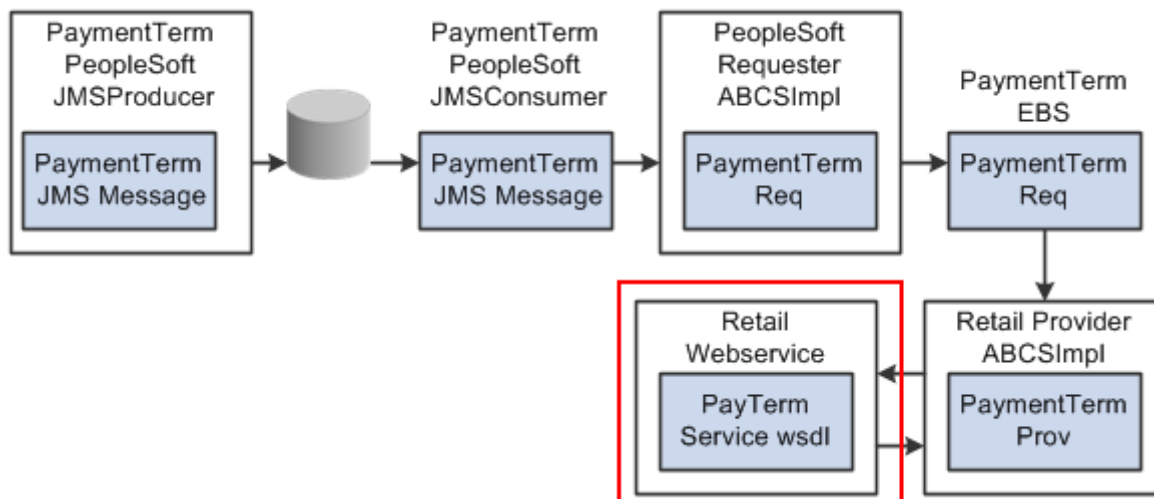


[UpdatePaymentTermRetailProvABCSImpl](#)

PayTermService

PayTermService accepts PaymentTermABM message types for operations createPayTermDesc, updateHeaderPayTermDesc, updatePayTermDesc, and createDetailPaytermDesc from SyncPaymentTermListRetailProvABCSImpl in case of synchronization of a bulk of updated payment terms data or UpdatePaymentTermRetailProvABCSImpl in case of 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 sending. While transforming the data, the DVM looksups the common status code to determine the RMS enabled flag value.

This diagram illustrates the relationship of the PayTermService with the other services in the integration flow:



[PayTermService](#)

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 Retail Merchandising System (RMS) and discusses:

- Supplier integration details
- PeopleSoft Payables interfaces
- RMS interfaces
- Data requirements
- Core AIA components
- Integration services

Supplier Integration Overview

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

Merchandise suppliers are suppliers of goods and services that the retailer sells to customers. PeopleSoft Payables and RMS require sharing of suppliers' information between them. RMS requires the supplier information for several key functions including creation and management of items and purchase orders. PeopleSoft Payables requires suppliers' information for supplier payment. For end-to-end business integration, same supplier instance and related information must be shared between these two systems.

Between the two systems, 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's information from PeopleSoft Payables to RMS through these integration flows:

- Load initial suppliers from PeopleSoft Payables to 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 RMS for a new instance (logical or physical) of 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 RMS: Enables the synchronization of incremental creation and updates of the current effective dated suppliers from PeopleSoft Payables to RMS.

For more information about supplier create and update operations, see *PeopleSoft Enterprise Source to Settle Common Information 9.0 PeopleBook*, "Maintaining Vendor Information"

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

Prerequisites

These are the prerequisites for this integration:

1. Initial load of the ORGANIZATION_ID cross-reference table using manual process. This is a manual post install step. 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 Chapter 7: Implementing Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials, section [Setting Up Cross-References for Oracle Retail IDs and PeopleSoft Entities](#).

2. Payment terms synchronization using the AIA process.
3. Freight terms synchronization using manual process.

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 RMS is not synchronized with PeopleSoft Payables.
3. The volume of payload (SupplierPartyList) which is handled by the process 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 Order from and Remit to addresses in detail.
5. The PeopleSoft system sends all the related information over to AIA.
For example, if an address is changed, then all the current effective locations having that address and all the suppliers linked to that location are included in the message. The same is true for the reverse. If the supplier information is changed, then all its locations and its addresses are sent.
6. The PeopleSoft system sends the common field for the DVM lookup.

No need exists for DVM lookup in the PeopleSoft ABCS. DVM lookup is done only in the RMS side Provider ABCS.

7. PeopleSoft suppliers have a record in the Oracle Retail language prior to the performing of the sync operation.

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

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

The 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 AIA layer changes its status as inactive. For the update operation, the AIA layer allows only approved suppliers.

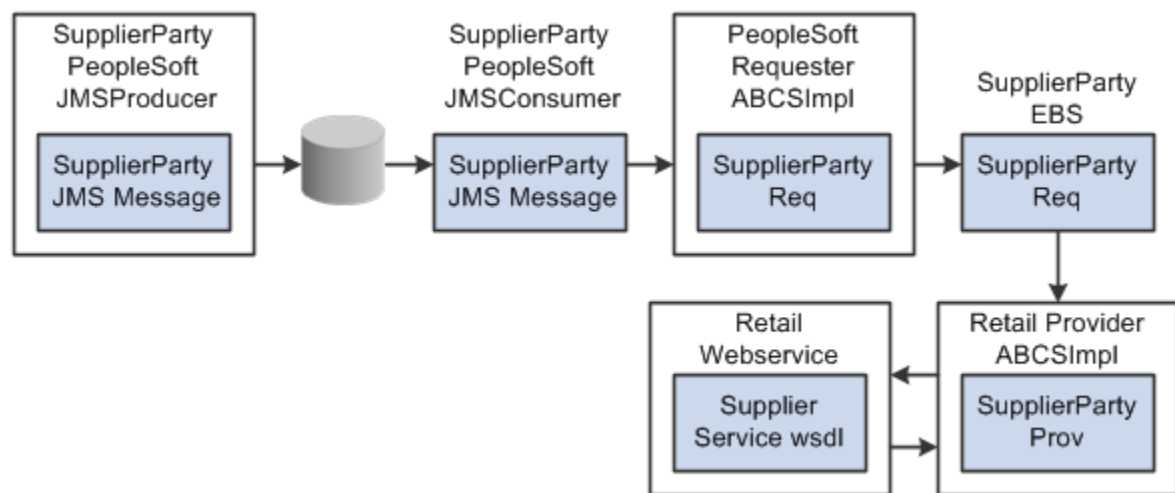
These are the constraints for the suppliers' integration:

1. The AIA services do not know the changes in the supplier unless PeopleSoft Payables notifies it.
2. The AIA integration solutions do not correct failure in the processing of messages in RMS. 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, there is a manual process to resubmit the message.

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

This diagram illustrates the supplier integration flow:



Supplier integration flow

Supplier Integration Details

These services are common to create, update, and sync supplier integration flows:

- SupplierPartyPeopleSoftJMSProducer

- SupplierPartyPeopleSoftJMSConsumer
- SupplierPartyEBS

These services are specific to sync supplier integration flow:

- SyncSupplierPartyListPeopleSoftReqABCImpl
- SyncSupplierPartyListRetailProvABCImpl

These services are specific to create supplier integration flow:

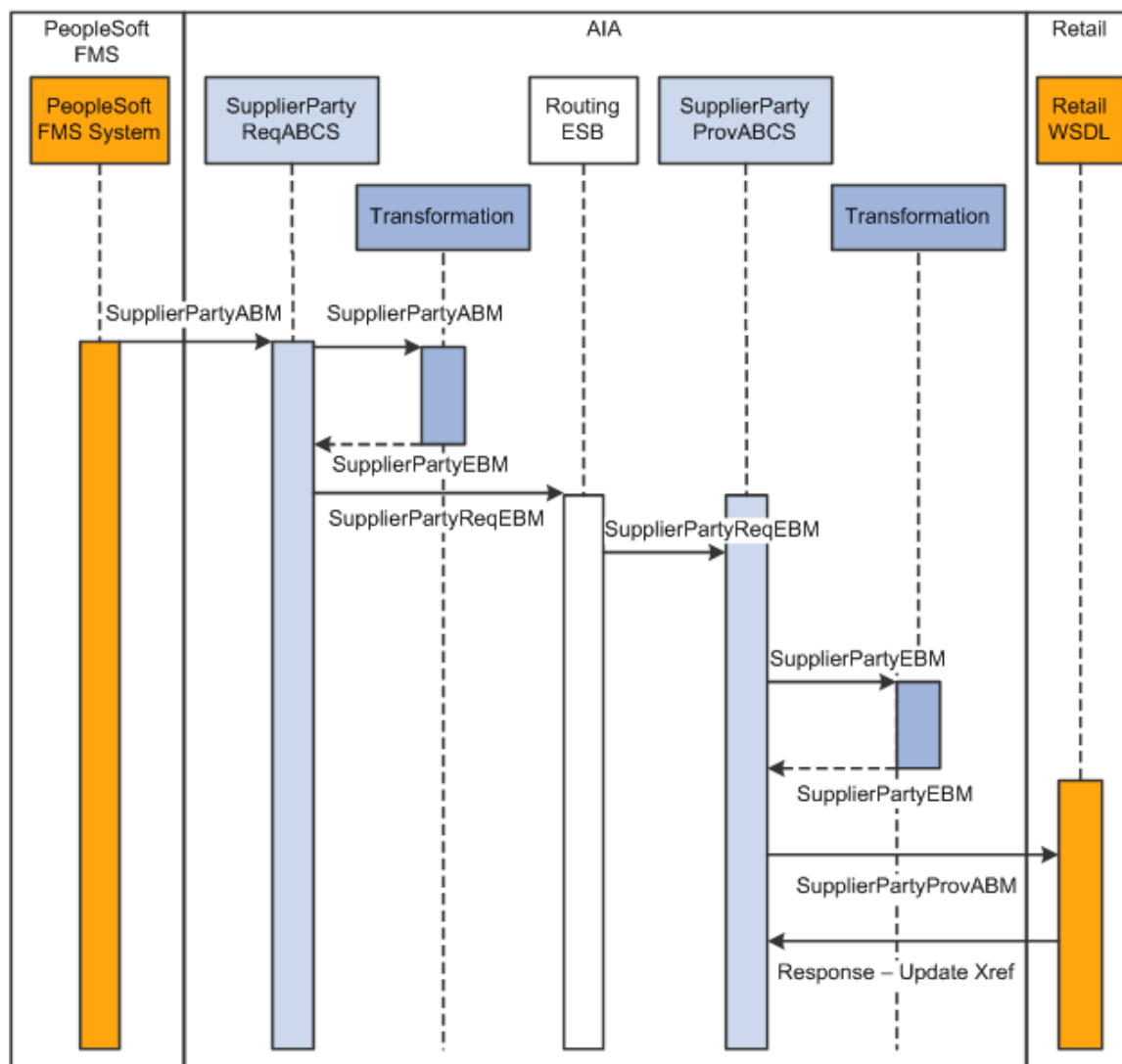
- CreateSupplierPartyPeopleSoftReqABCImpl
- CreateSupplierPartyRetailProvABCImpl

These two services are specific to update supplier integration flow.

- UpdateSupplierPartyRetailProvABCImpl
- UpdateSupplierPartyPeopleSoftReqABCImpl

Sequence Diagram

This sequence diagram illustrates the supplier information integration flow:



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 invokes the SupplierPartyPeopleSoftJMSProducer service with SyncSupplierPartyListPSFTABM/ CreateSupplierPartyPSFTABM/ UpdateSupplierPartyPSFTABM.

This service puts this ABM in the AIA_SUPPLIERPARTYJMSQUEUE with JMSCorrelationID ID set to Sync/Create/Update.
2. The SupplierPartyPeopleSoftJMSConsumer service listens on the AIA_SupplierPartyJMSQueue and invokes SyncSupplierPartyListPeopleSoftReqABCImpl in case of SyncSupplierPartyListPSFTABM, CreateSupplierPartyPeopleSoftReqABCImpl in case of CreateSupplierPartyPSFTABM, and UpdateSupplierPartyPeopleSoftReqABCImpl in case of UpdateSupplierPartyPSFTABM.

3. The SyncSupplierPartyListPeopleSoftReqABCSEImpl transforms the SupplierPartyListABM message into SupplierPartyListEBM.

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

4. The CreateSupplierPartyPeopleSoftReqABCSEImpl transforms the SupplierPartyABM message into SupplierPartyEBM.

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

5. The UpdateSupplierPartyPeopleSoftReqABCSEImpl transforms the SupplierPartyABM message into SupplierPartyEBM.

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

6. SupplierPartyEBS routes the SupplierPartyListEBM to Retail provider ABCS implementation. It also provides CAVS routing.

7. SyncSupplierPartyListRetailProvABCSEImpl transforms SupplierPartyListEBM to Retail supplier message ABM.

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

8. CreateSupplierPartyRetailProvABCSEImpl transforms SupplierPartyEBM to Retail supplier message ABM.

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

9. UpdateSupplierPartyRetailProvABCSEImpl transforms SupplierPartyEBM to Retail supplier message ABM.

Transformation applies the DVM and invokes update web service from 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 an EBM. PeopleSoft ABM does not have some details such as the EBM header and the 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.

Multiple Language Support

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

The PeopleSoft system supports several languages per instance based on the 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 in the PeopleSoft system. For example, the data field for the supplier name has a French entry from a user with French UI preference and an English entry from a user with English UI preference.

Because Oracle Retail supports only one primary language per instance, the 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 look up based on the common EBM Header. If the DVM language code look up 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 supplier, AIA does not filter the message from PeopleSoft. AIA sends the message in the same language in which it was created or updated, even this language is not the primary language of Oracle Retail instance.

AIA on the 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.

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`

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*, Volume 2, “Web Services Overview”

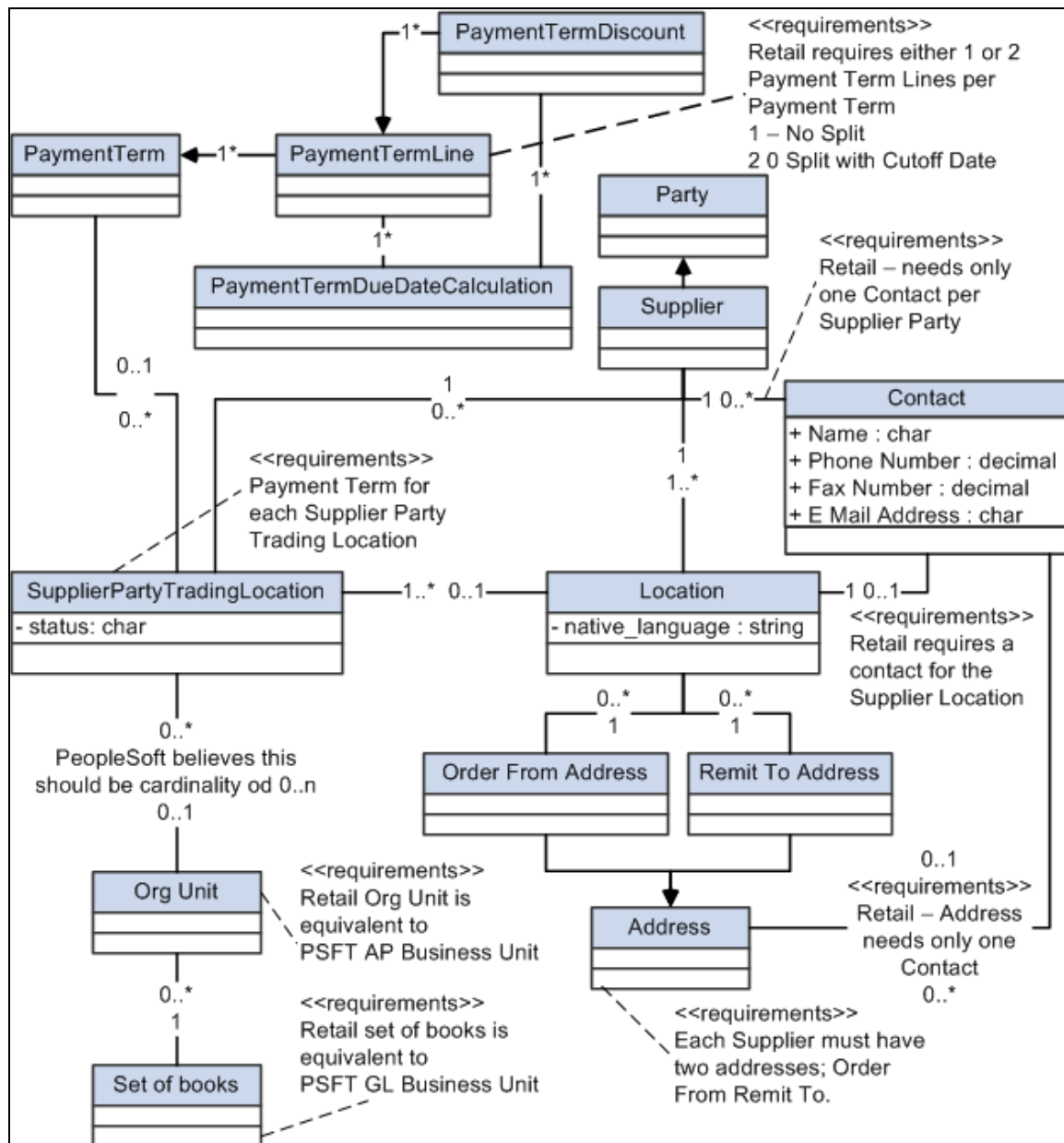
Data Requirements

The process integration for supplier information requires these data:

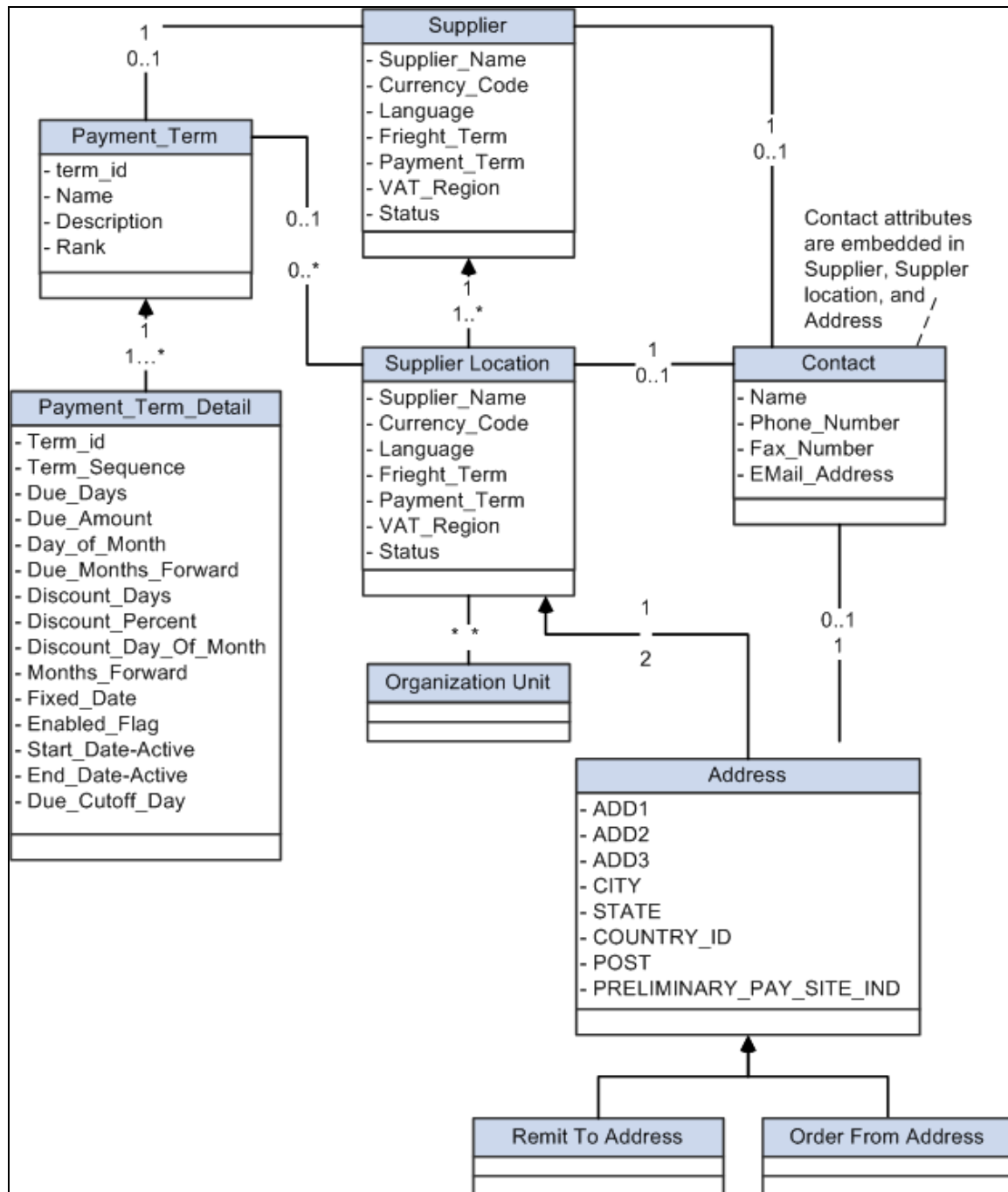
- Suppliers must be assigned to at least one GL business unit (set of books)
- For this integration, PeopleSoft suppliers must be created with These status attributes:
 - Approved.
 - Classification of supplier. Other type of vendors such as attorneys, employees, and HCM are not synchronized with Oracle Retail.
 - The Open for ordering option is selected. If the supplier is approved but is not open for ordering, the AIA layer on the 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.

Class Diagram

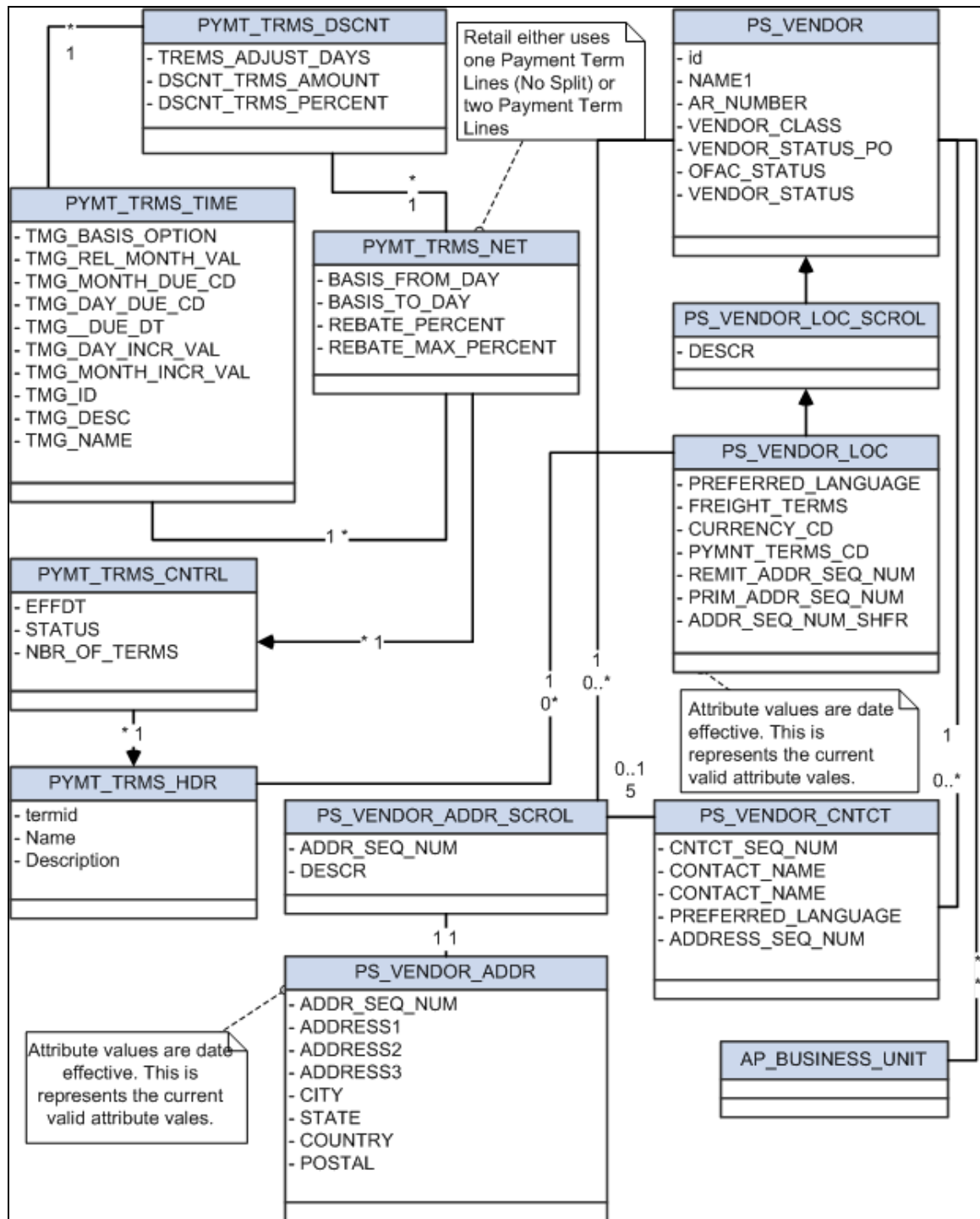
These are class diagrams for supplier integration:



Class diagram 1 of 3



Class diagram 2 of 3



Class diagram 3 of 3

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:
[http://\[HOST:PORT\]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/](http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/).

The core EBS WSDL files can be located by EBO within this parent folder:
[http://\[HOST:PORT\]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/](http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/).

For detailed documentation of individual EBOs, click the EBO name link on the Integration Scenario Summary page in the Oracle AIA Console. You can also use the Integration Scenario Summary page to search for and view integration scenarios that use a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” “Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

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

For more information, see *Oracle Application Integration Architecture - Foundation Pack Integration Developer's Guide*, “Extensibility for Oracle AIA Artifacts,” Extending EBOs

Integration Services

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

- SupplierPartyPeopleSoftJMSProducer
- SupplierPartyPeopleSoftJMSConsumer
- SyncSupplierPartyListPeopleSoftReqABCSImpl
- CreateSupplierPartyPeopleSoftReqABCSImpl
- UpdateSupplierPartyPeopleSoftReqABCSImpl
- SupplierPartyEBS
- SyncSupplierPartyListRetailProvABCSImpl
- CreateSupplierPartyRetailProvABCSImpl
- UpdateSupplierPartyRetailProvABCSImpl

You can use the Integration Scenario Summary page in the Oracle AIA Console to search for and view integration scenarios that use a particular ABC service.

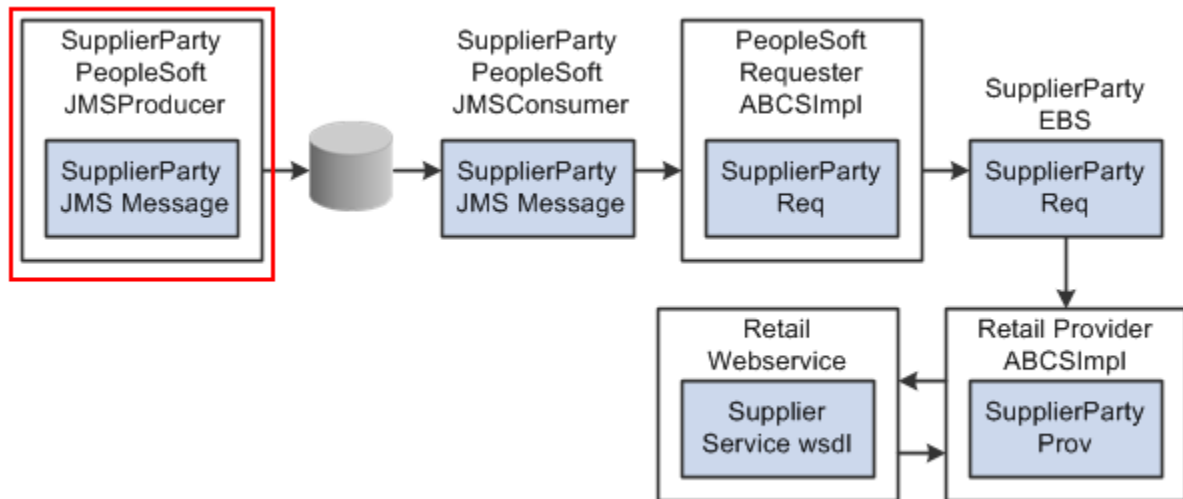
For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios

SupplierPartyPeopleSoftJMSProducer

PeopleSoft Payables invokes the SupplierPartyPeopleSoftJMSProducer service for the initial load and incremental create and update operation. The same service can support the bulk update of suppliers from PeopleSoft to RMS.

This is a BPEL process. This process receives three different types of message in CreateSupplierParty, UpdateSupplierParty, and SyncSupplierPartyList with three different operations. It publishes the message in the queue with the help of JMSAdapter. JMSHeader is set before enqueue of the message. Values for the JMSHeader correlation are Sync, Create, and Update. The message consumers pick up the message based on the header value.

This diagram illustrates the relationship of the SupplierPartyPeopleSoftJMSProducer with the other services in the integration flow:



SupplierPartyPeopleSoftJMSProducer

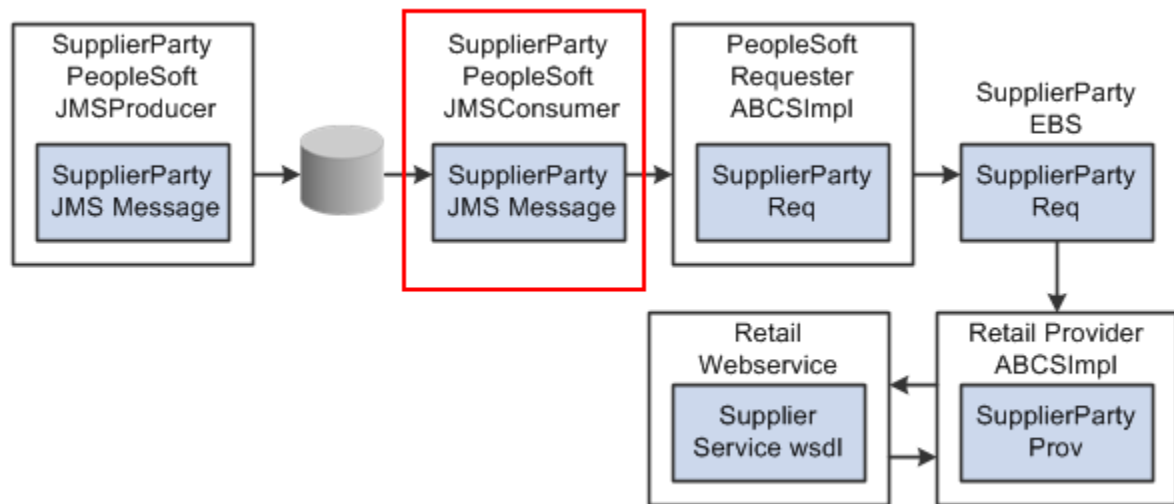
SupplierPartyPeopleSoftJMSConsumer

SupplierPartyPeopleSoftJMSConsumer is an ESB. It has three JMSAdapter services for dequeue of the message from the queue. After dequeuing the messages, it calls for the respective PeopleSoftReqABCSImpl (Create, Update, or Sync).

This service has These JMS adapters:

- SyncSupplierPartyPeopleSoftJMSConsumer
- CreateSupplierPartyPeopleSoftJMSConsumer
- UpdateSupplierPartyPeopleSoftJMSConsumer

This diagram illustrates the relationship of the SupplierPartyPeopleSoftJMSConsumer with the other services in the integration flow:



SupplierPartyPeopleSoftJMSConsumer

SyncSupplierPartyListPeopleSoftReqABCSImpl

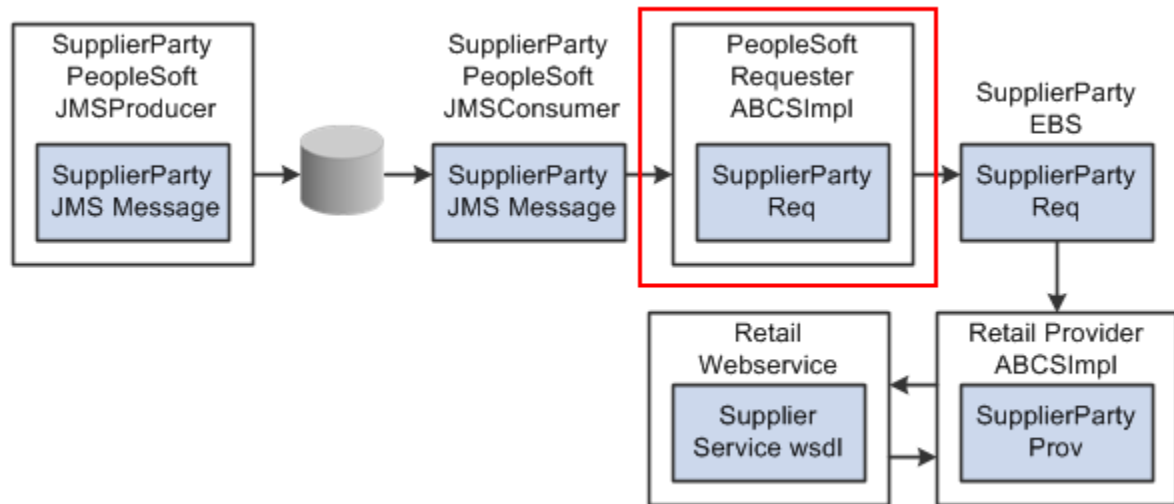
SupplierPartyPeopleSoftJMSConsumer invokes the SyncSupplierPartyListPeopleSoftReqABCSImpl service for the list of existing suppliers during the initial integration with other applications.

This is a single operation service that has SupplierPartyEBS as a partner service. This 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 an XREF for suppliers and suppliers' addresses.
- Transforms SyncSupplierPartyListReqMsg to SyncSupplierPartyListEBM. While it is transforming ABM to EBM, cross-references are looked up for SUPPLIERPARTY_ID, SUPPLIERPARTY_LOCATION_ID, and SUPPLIERPARTY_ADDRESS_ID.
- Sends SyncSupplierPartyListEBM messages as an input to the SyncSupplierPartyList operation in the SupplierPartyEBS service.

This diagram illustrates the relationship of the SyncSupplierPartyListPeopleSoftReqABCSImpl with the other services in the integration flow:



SyncSupplierPartyListPeopleSoftReqABCSImpl

CreateSupplierPartyPeopleSoftReqABCSImpl

This 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.

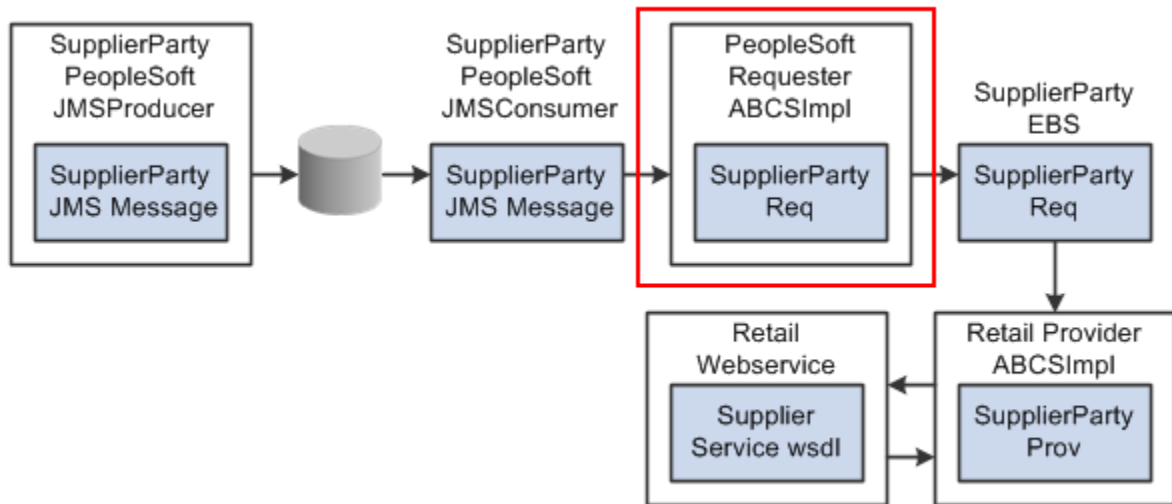
This message contains an XREF for suppliers and suppliers' addresses.

- Transforms CreateSupplierPartyReqMsg to CreateSupplierPartyEBM.

While it is transforming ABM to EBM, cross-references are looked up 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.

This diagram illustrates the relationship of the CreateSupplierPartyPeopleSoftReqABCSImpl with the other services in the integration flow:



CreateSupplierPartyPeopleSoftReqABCSImpl

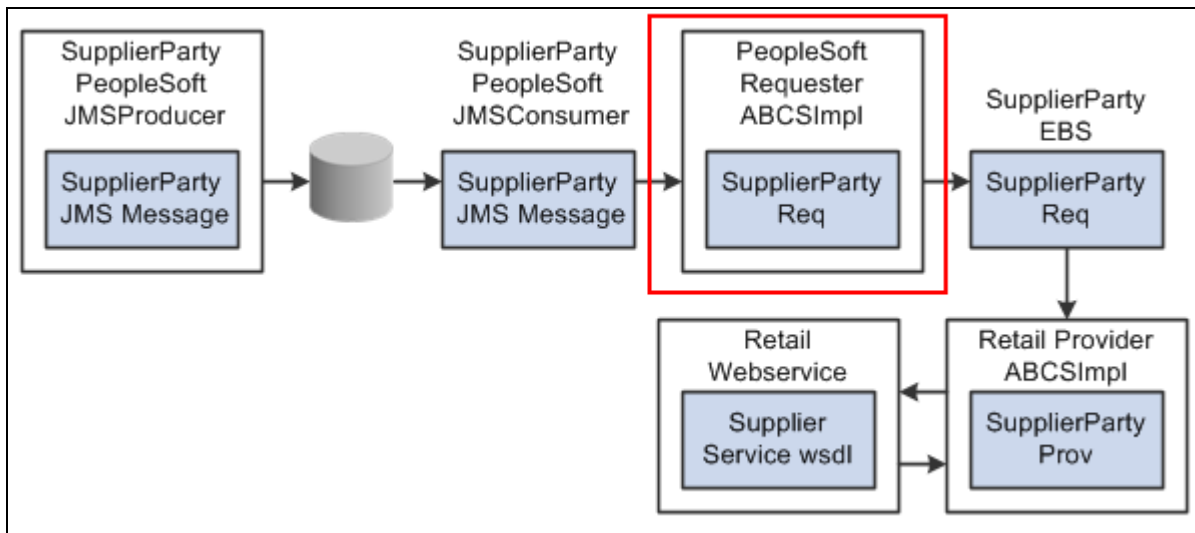
UpdateSupplierPartyPeopleSoftReqABCSImpl

This is a single operation service. It 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 SupplierPartyPeopleSoftJMSConsumer.

The process performs these actions:

- Accepts the UpdateSupplierPartyReqMsg message from PeopleSoft Payables.
This message contains an XREF for suppliers and suppliers' addresses.
- Transforms UpdateSupplierPartyReqMsg to UpdateSupplierPartyEBM.
While it is transforming ABM to EBM, a cross-reference is looked up 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.

This diagram illustrates the relationship of the UpdateSupplierPartyPeopleSoftReqABCSImpl with the other services in the integration flow:

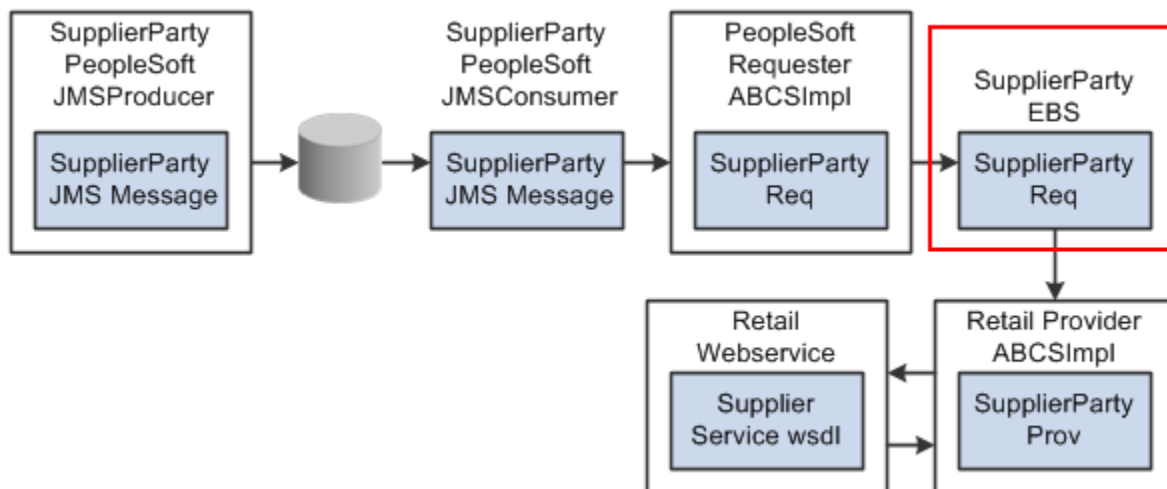


UpdateSupplierPartyPeopleSoftReqABCSImpl

SupplierPartyEBS

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

This diagram illustrates the relationship of the SupplierPartyEBS with the other services in the integration flow:



SupplierPartyEBS

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

- SyncSupplierPartyList
- CreateSupplierParty

- UpdateSupplierParty

For more information about this EBS, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Designing and Developing EBSs” and *Oracle Application Integration Architecture – Foundation Pack: Concepts and Technologies Guide*, “Understanding EBSs”

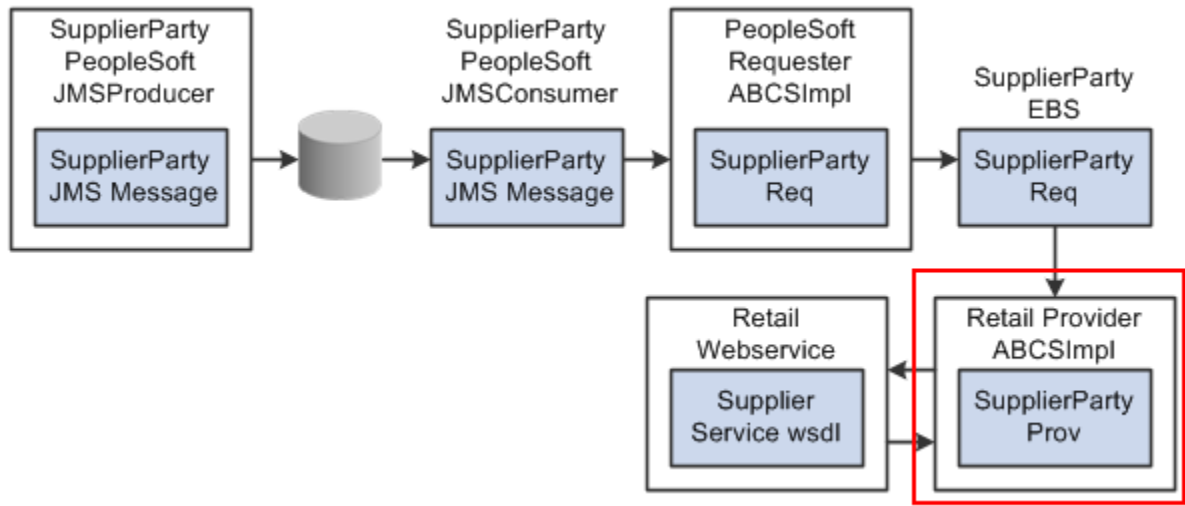
SyncSupplierPartyListRetailProvABCSImpl

SyncSupplierPartyListRetailProvABCSImpl 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.

This diagram illustrates the relationship of the SyncSupplierPartyListRetailProvABCSImpl with the other services in the integration flow:



SyncSupplierPartyListRetailProvABCSImpl

CreateSupplierPartyRetailProvABCSImpl

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

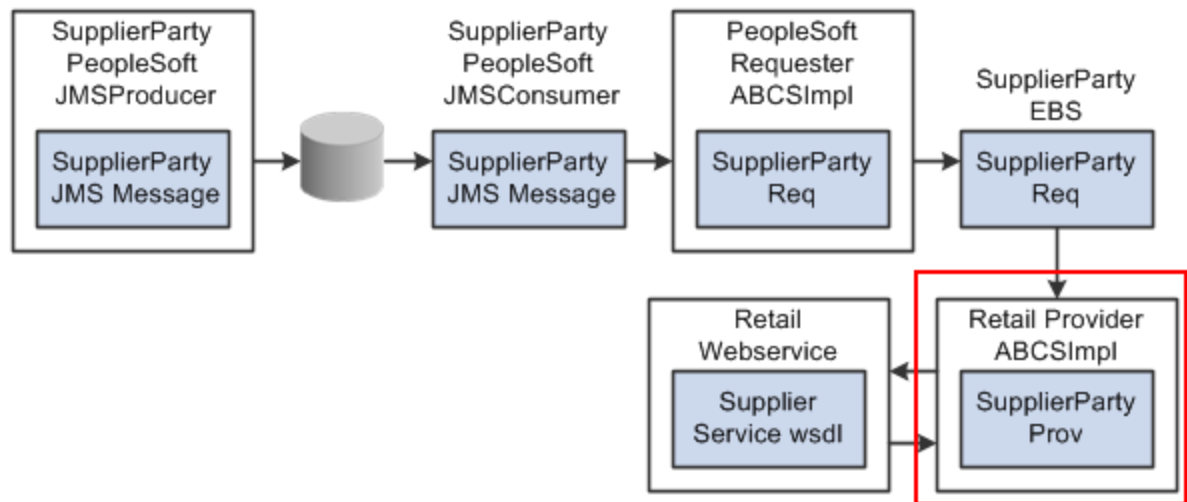
This process performs these actions:

- 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 RMS web service is invoked for create.

- Updates the cross-references for Oracle Retail after the Oracle Retail web service call.

This diagram illustrates the relationship of the CreateSupplierPartyRetailProvABCSImpl with the other services in the integration flow:



CreateSupplierPartyRetailProvABCSImpl

UpdateSupplierPartyRetailProvABCSImpl

UpdateSupplierPartyRetailProvABCSImpl is a single operation service that receives input from SupplierPartyEBS. It invokes the Update web service in 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.

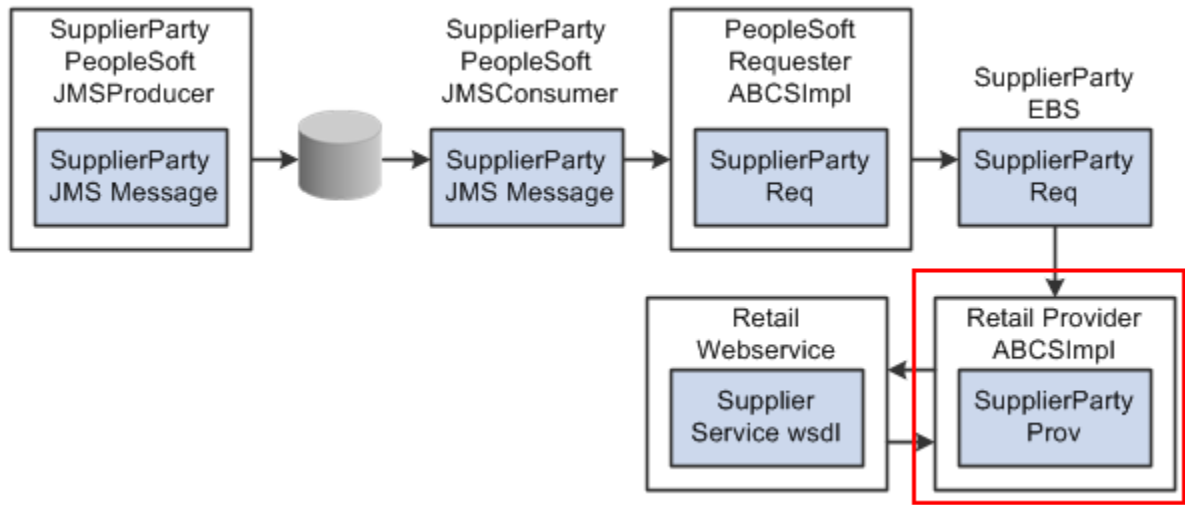
An UpdateSupplierPartyEBM message contains these details:

- 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.

This diagram illustrates the relationship of the UpdateSupplierPartyRetailProvABCSImpl with the other services in the integration flow:



UpdateSupplierPartyRetailProvABCSImpl

Chapter 3: Chart of Accounts Validation Integration Flow

This chapter describes how to request the chart of accounts combination validation and discusses:

- Oracle Retail interfaces
- PeopleSoft GL interfaces
- Data requirements
- Core Application Integration Architecture (AIA) components
- Integration services

Requesting Chart of Accounts Combination Validation

PeopleSoft General Ledger (GL) is the system of record for chart of accounts segment combinations. Chart of accounts ChartFields (segments) combinations are set up manually in Oracle Retail Merchandising System (RMS), Retail Invoice Matching (ReIM), and Retail Sales Audit (ReSA). This manual setup enables assignment of transaction data in sales audit, stock ledger, and invoice match to specific account codes. 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 (RMS, ReIM, or ReSA)
- Set of books (GL business unit)
- Segment or ChartField combination values

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

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

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 (RMS, ReSA, or ReIM).

2. PeopleSoft GL accepts all the AIA Common values in the provider ABM.

Therefore, no DVM lookups are required on the PeopleSoft provider side. The DVM values are entered in AIA and PeopleSoft.

3. Multiple sets of GL accounts can be sent from Oracle Retail to the PeopleSoft validation web service.

Chart of Accounts Combination Validation Integration Flow

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

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

This diagram illustrates the chart of accounts combination validation integration flow:

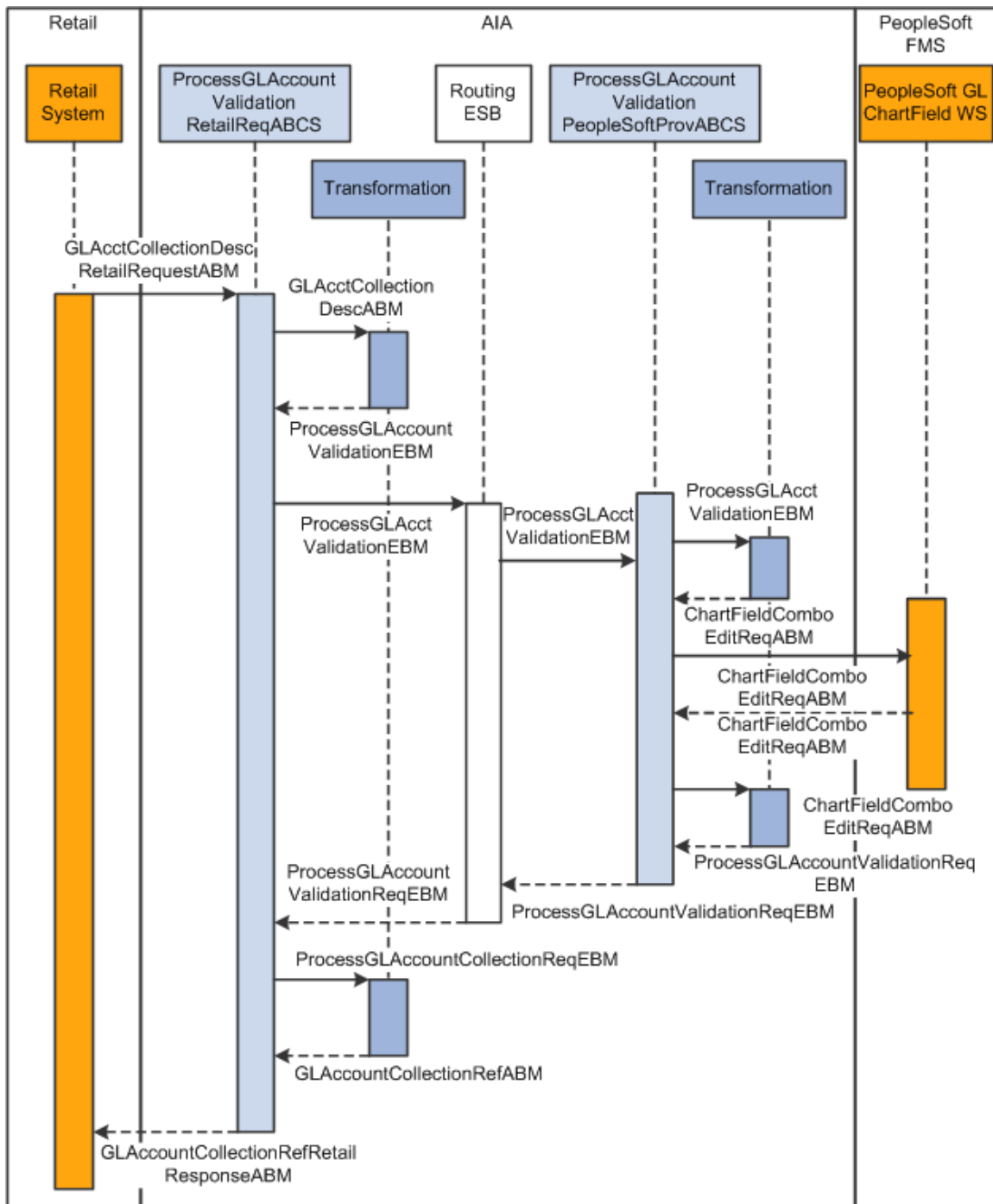


Chart of accounts combination validation sequence diagram

When the chart of accounts validation process is initiated:

1. Oracle Retail calls the **ProcessGLAccountValidationRetailReqABCSImpl** service with the **ProcessGLAccountValidationRetailABM** (**GLAcctDesc.xsd**) whenever a GL account is required to be validated.

The ProcessGLAccountValidationRetailReqABCSImpl service transforms the ProcessGLAccountValidationRetailABM into the ProcessGLAccountValidationEBM and then invokes the ChartOfAccountsEBS. When it returns from the ChartOfAccountsEBS, it transforms the ProcessGLAccountValidationResponseEBM to the ProcessGLAccountValidationResponseRetailABM (GLAcctRef.xsd) and sends the response back to Oracle Retail.

2. ChartOfAccountsEBS invokes ProcessGLAccountValidationPeopleSoftProvABCSImpl or Composite Application Validation System (CAVS) based on the routing rule.
3. ProcessGLAccountValidationPeopleSoftProvABCSImpl invokes the PeopleSoft GL_CHARTFIELD service with ProcessGLAccountValidationEBM for validating the GL account.

Then ProcessGLAccountValidationPeopleSoftProvABCSImpl returns ProcessGLAccountValidationResponseEBM to ChartOfAccountsEBS.

Oracle Retail Interfaces

Retail Outbound Interactions:

- Name: ProcessGLAccountValidation: Oracle Retail (RMS, RESA, or 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.

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 9.0 PeopleBook*, “Integrating and Transferring Information Among Applications”

Data Requirements

The segment combinations in the GL ACCOUNT MAINTENANCE (Sales Audit), GL CROSS REFERENCE (RMS), and GL CROSS REFERENCE (ReIM) forms must be entered manually.

Core AIA Components

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

- ChartOfAccountsEBO
- ChartOfAccountsEBM
- ChartOfAccountsEBS

The core EBO and EBM XSD files can be located by EBO within this parent folder:
http://[HOST:PORT]/AIAComponents/EnterpriseObjectLibrary/Core/EBO/.

The core EBS WSDL files can be located by EBO within this parent folder:
http://[HOST:PORT]/AIAComponents/EnterpriseBusinessServiceLibrary/Core/EBO/.

For detailed documentation of individual EBOs, click the EBO Name link on the Integration Scenario Summary page in the Oracle AIA Console. You can also use the Integration Scenario Summary page to search for and view integration scenarios that use a particular EBO or EBS.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” “Using the BSR UI to View Integration Scenarios,” Viewing Integration Scenarios

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

For more information, see *Oracle Application Integration Architecture - Foundation Pack Integration Developer’s Guide*, “Extensibility for Oracle AIA Artifacts,” Extending EBOs

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, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Using the BSR UI to View Integration Scenarios

ProcessGLAccountValidationRetailReqABCSImpl

Oracle Retail (RMS, ReSA, or ReIM) invokes this requestor 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 sends it to the calling application.

In this requestor 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 CHARTOFACCOUNTS_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 CHARTOFACCOUNTS_ACCOUNTSTATUS DVM table is used to transform the AIA Common/PSFT value to the Oracle Retail value.

ProcessGLAccountValidationRetailReqABCSImplExt is a partner link service. This 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.

ChartOfAccountsEBS

ChartOfAccountsEBS calls ProcessGLAccountValidationRetailReqABCSImpl. 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 ProcessGLAccountValidationRetailReqABCSImpl service.

For more information about this EBS, see *Oracle Application Integration Architecture – Foundation Pack: Integration Developer’s Guide*, “Designing and Developing EBSs” and *Oracle Application Integration Architecture – Foundation Pack: Concepts and Technologies Guide*, “Understanding EBSs.”

ProcessGLAccountValidationPeopleSoftProvABCSImpl

The ProcessGLAccountValidationPeopleSoftProvABCSImpl service receives ProcessGLAccountValidationEBM as an input from ChartOfAccountsEBS. This is a single operation service. This service expects ProcessGLAccountValidationEBM and publishes ProcessGLAccountValidationEBM to PeopleSoft Code Combination Validate 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 AIA common format.

When the ChartOfAccountsEBS service invokes the ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc service:

- ChartOfAccountsEBS calls ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc with ProcessGLAccountValidationEBM. EBM_Header variable is assigned with the header information of ProcessGLAccountValidationEBM.
- The ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc 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 AIA Common values.
- ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc's PreProcessABM is called with ProcessGLAccountValidationEBM based on a parameter in the AIAConfig file.
- ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc's 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.
- ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc's PostProcessEBM is called with ProcessGLAccountValidationResponseEBM based on a parameter in the AIAConfig file.
- ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc returns ProcessGLAccountValidationResponseEBM to the caller ChartOfAccountsEBS.

ProcessGLAccountValidationPeopleSoftProvABCSEBMSvc is a partner link service. This 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.

Chapter 4: 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 (ReSA), Retail Merchandising System (RMS), and Retail Invoice Matching (ReIM) to PeopleSoft General Ledger (GL) and discusses how to:

- Perform setup tasks
- Configure and generate data in Oracle Retail applications
- Pick up and transform the data
- Configure and run the process integration for accounting entries

Process Integration for Inventory Valuation and Revenue Recognition Accounting Entries Overview

The process integration for accounting entries enables you to record the financial impact of changes to sellable store and warehouse inventory. It also records the financial impact 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 GL through accounting entries. Similarly the system sends the prepaid invoice reversals and write-offs of aged receipts PeopleSoft GL. RMS, ReIM, and ReSA are the source for accounting entry. After successful completion of data transfer, the system deletes the data from the source interface tables.

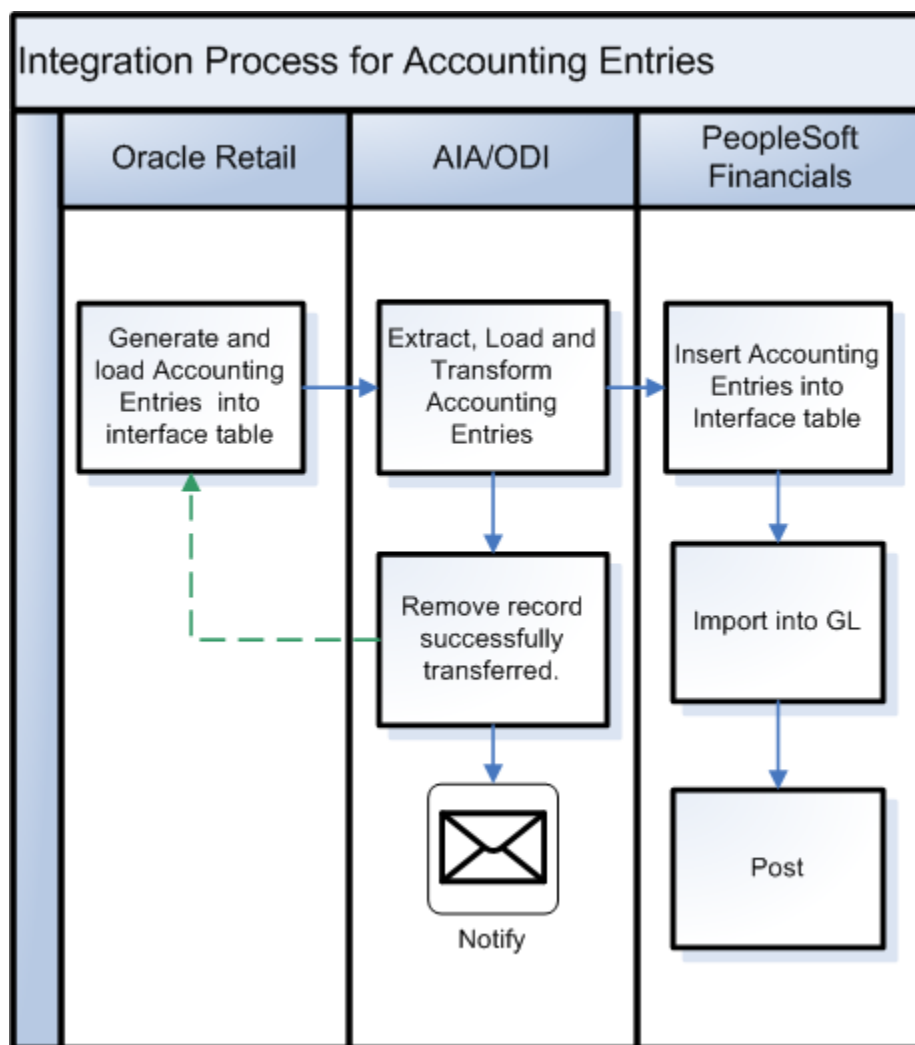
This integration uses the Oracle Data Integrator (ODI) application to transfer the data from the Oracle Retail applications to the PeopleSoft GL application.

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

- Post stock ledger from RMS to PeopleSoft GL.
- Post channel sales, cash, and deposits from ReSA to PeopleSoft GL.
- Post write-offs (aged receipts, not invoiced to ledger) from RMS to PeopleSoft GL.
- Post prepaid invoice reversals after receipt from ReIM to PeopleSoft GL.

Business Process Flow for the Integration of Accounting Entries

This flowchart illustrates the process integration for accounting entries:



Business process flow for the integration of accounting entries

The overall process includes:

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.

Prerequisites

Before performing this process integration, ensure that:

1. These Domain Value Maps (DVMs) have been entered on the Enterprise Service Bus (ESB) console, and exported, also setup in the PeopleSoft system:
 - a. BUSINESS_UNIT
 - b. CURRENCY_CODE
2. These views have been created in the Oracle Retail database:
 - a. FIF_STG_GL_DATA_VW
 - b. IM_FINANCIALS_STG_VW
3. The AIAConfigurationProperties.XML file is set up.

Facts and Constraints

Oracle Retail:

1. RMS, ReSA, and ReIM are the source of the accounting entry flow.
2. Oracle Retail populates one of the staging tables: STG_FIF_GL_DATA or IM_FINANCIALS_STAGE (depending on whether the RMS or 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; however, this 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 decimal and Oracle Retail allows 4 digits. So some data may be rounded while transferring from Oracle Retail to PeopleSoft GL.

Performing Setup Tasks

No specific setup required for PeopleSoft GL and Oracle Retail.

Perform these setup tasks specific to the integration layer.

Setup Tasks Specific to the Integration Layer

You must complete these tasks in the integration layer:

- Set up the required DVMs on ESB console, export from ESB, and also set up them in the PeopleSoft system.
- Create Views in the Oracle Retail database: (IM_FINANCIALS_STG_VW and FIF_STG_GL_DATA_VW).
- Set up the AIAConfigurationProperties.XML file. Set up only the email properties and data polling time.

Configuring and Generating Data in Oracle Retail

This section discusses:

- Configuring the data in Oracle Retail
- Generating the data in Oracle Retail

Configuring the Data in Oracle Retail

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

- Set up general ledger (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”

Generating the Data in Oracle Retail

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
- FIFGLDN2.PC
- FIFGLDN3.PC

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

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

1. Performs any resolution actions (for example, initiate 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. Once all of the resolution documents for a matched invoice are built, and all of the RCA/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.

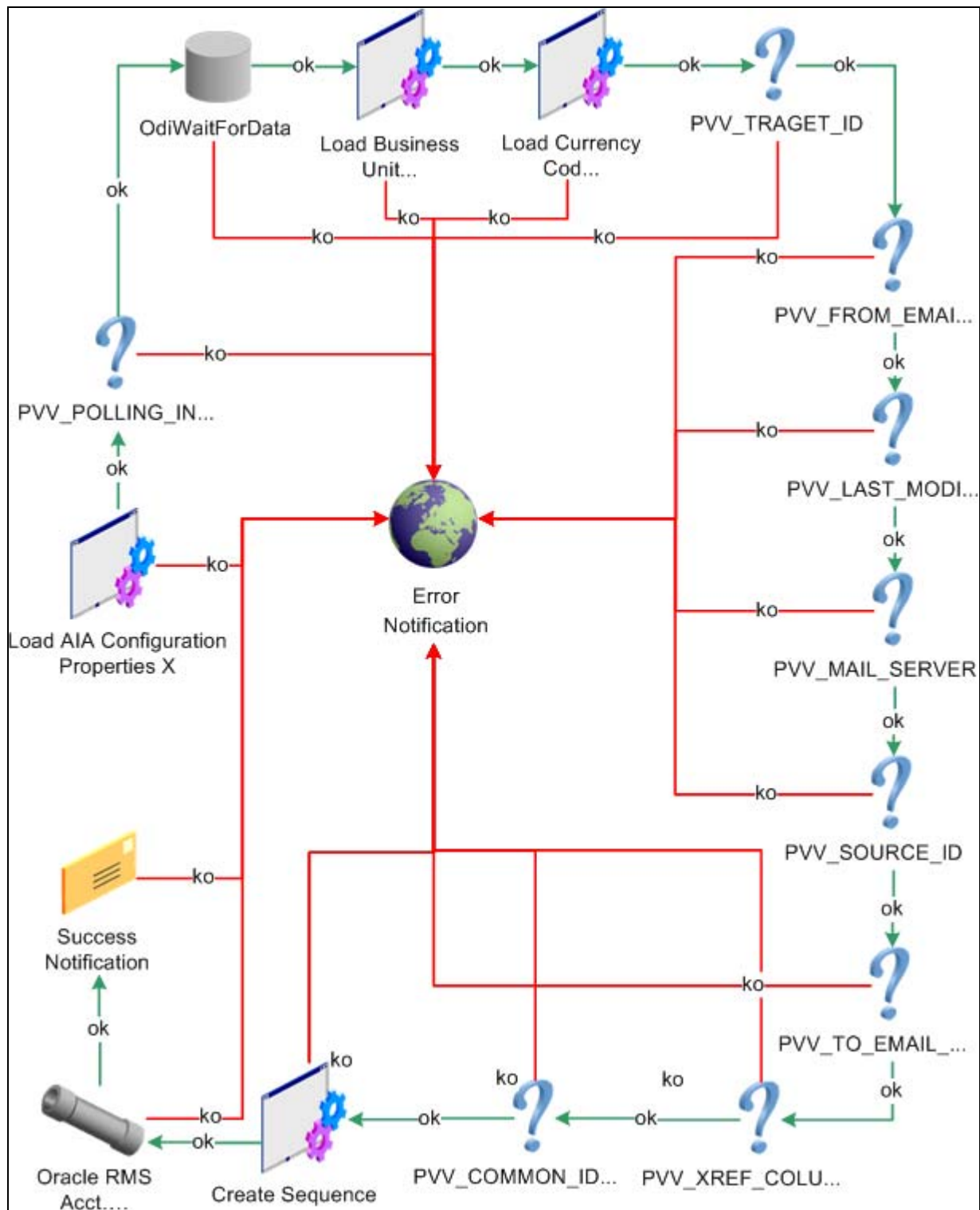
Once all of the transactions have been written, the process switches the status of the current invoices or documents to “Posted”, and then moves on to the next invoice or document. If a segment look-up fails, the failed record is written to a financials error table.

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

Picking Up and Transforming the Data

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

This flowchart illustrates the process:



Accounting entries load process

These tasks are performed as a 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 RMS or 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 already 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.

The system deletes the data in the STG_FIF_GL_DATA and IM_FINANCIALS_STG tables.

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.

Configuring and Running the Process Integration for Accounting Entries

This section discusses how to:

- Configure the process integration for accounting entries
- Set up a schedule
- Run the process integration for accounting entries

Configuring the Process Integration for Accounting Entries

After you have installed the process integration for accounting entries, you need to configure several parameters. Open the AIAConfigurationProperties.XML file that is available in the <AIA_HOME>/config folder.

Note: Whenever the AIAConfigurationProperties.XML file is updated, the file must be reloaded for updates to be reflected in the applications or services that use the updated properties. You can perform this reload by clicking the Reload button on the Configuration page in the Oracle AIA Console. Alternatively, you can perform the reload by rebooting the server.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Loading Oracle AIA Configuration File Updates

Setting Up AIAConfigurationProperties.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 sender's email address for email notification.

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

For the MailServerName property, configure the valid mail server which will be used to send email notifications.

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

If desired, update the PollingInterval property:

```
<Property name="PollingInterval">60000</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>
```

Setting Up and Exporting DVMs

During the installation process, the Oracle Universal Installer prompted for a location to export the DVMs. After setting up the DVMs, export these DVMs from ESB to that location:

- BUSINESS_UNIT
- CURRENCY_CODE

Set up and export the CHARTOFACCOUNTS_GLELEMENT DVM to
<AIA_HOME>/PIPS/Core/DIS/RetailToPSFTFin/CreateRetailViewSQL>.

For more information about how to create DVMs, see *Oracle Enterprise Service Bus Developer's Guide*, "Creating and Populating Domain-Value Maps"

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.

Creation of the views in the Oracle Retail database includes these steps:

1. Set up the cross-reference between segments and ChartFields.

- a. Update the DVM: In the ESB server, update the DVM CHARTOFACCOUNTS_GLELEMENT. Map the Oracle Retail segments with the corresponding PeopleSoft ChartFields.
 - b. Export CHARTOFACCOUNTS_GLELEMENT DVM into the <AIA_HOME>\PIPS\Core\DIS\RetailToPSFTFin\CreateRetailViewSQL folder.
2. Run the Java program to generate the SQL scripts.
 - a. From the command line, navigate to the CreateRetailViewSQL folder.
 - b. Ensure that the Java version is JDK 1.6.
 - c. Call the ANT script CreateRetailViewSQL.xml.

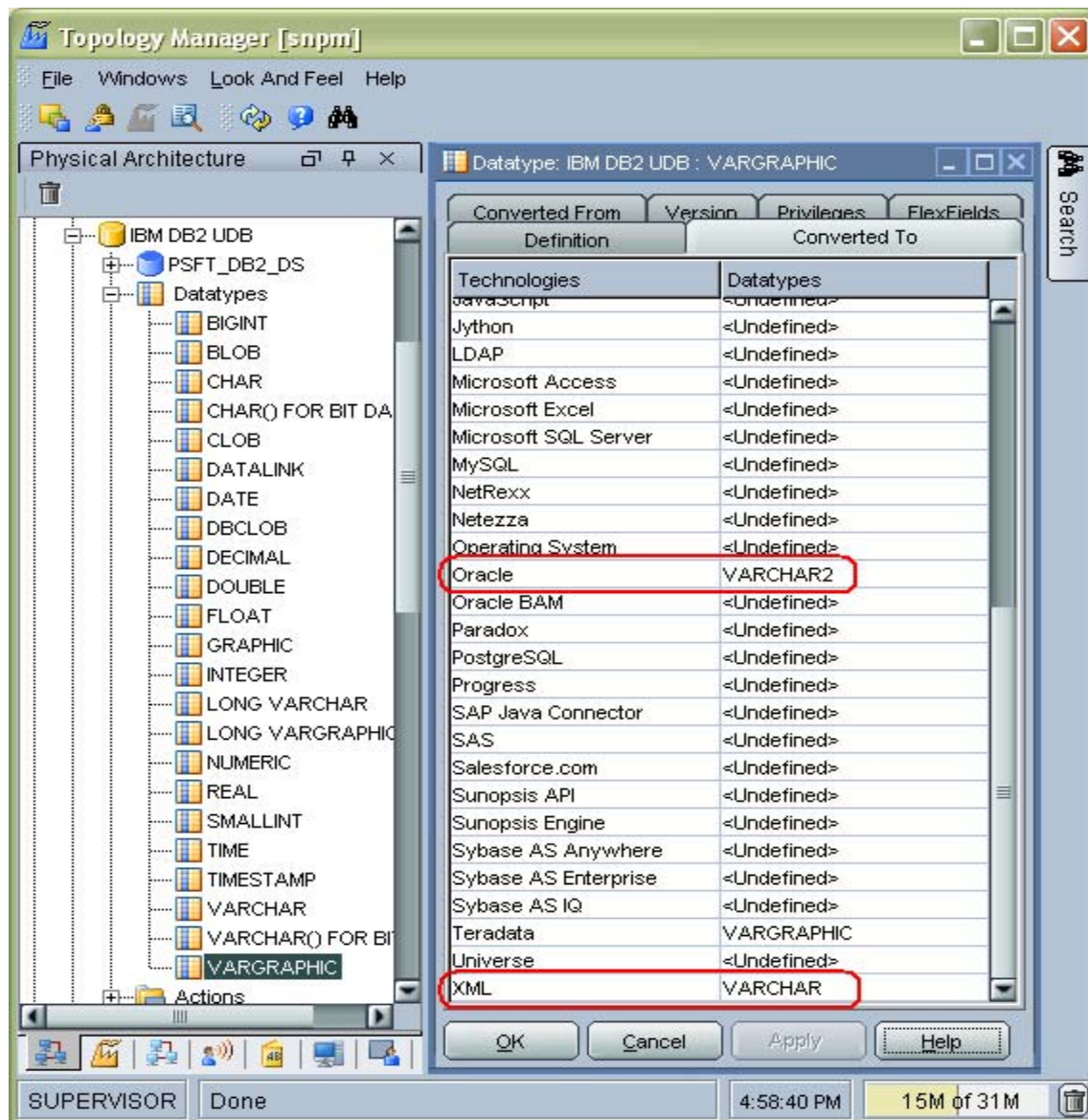
This ANT script generates these SQL scripts:
STG_FIF_GL_DATA_VW.sql
IM_FINANCIALS_STG_VW.sql
IM_AP_STAGE_DETAIL_VW.sql
3. Run these SQL scripts into the Oracle Retail database to create the views:
STG_FIF_GL_DATA_VW.sql, IM_FINANCIALS_STG_VW.sql, and IM_AP_STAGE_DETAIL_VW.sql.

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 PeopleSoft 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 this image, before generating the scenario:



Set up the Datatype conversion to VARGRAPHIC

For each package, users need to generate a scenario and then they 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.

Running the Process Integration for Accounting Entries

This process runs based on the schedules created for it. The integration between Oracle Retail and PeopleSoft GL can be scheduled 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

Chapter 5: 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 (ReIM) and PeopleSoft Payables and discusses how to:

- Perform setup tasks
- Configure and generate data in Oracle Retail
- Pick up and transform the data
- Configure and run the process integration for Retail Merchandise Procure to Pay

Process Integration for Retail Merchandise Procure to Pay Overview

ReIM is the source of matched invoices, credit notes, debit memos, and rebates. PeopleSoft Payables requires these details for payment to the 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 prior to 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 ReIM to PeopleSoft Payables and General Ledger. Other accounting transactions are generated from ReIM to write off aged receipts that were never invoiced and to post accounting distribution for manually paid or prepaid invoices after receipt.

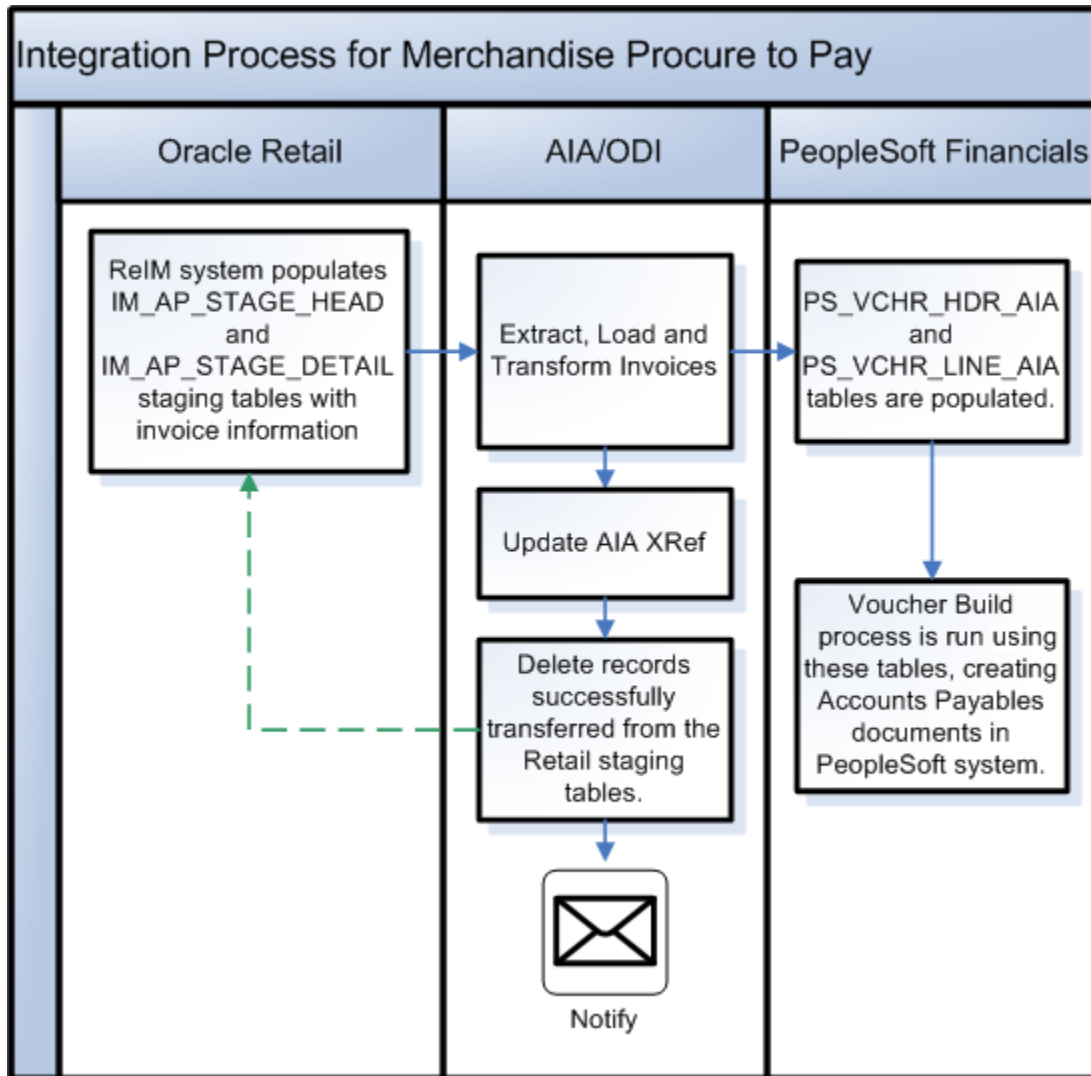
This integration uses the Oracle Data Integrator (ODI) application to transfer the data from ReIM to PeopleSoft Payables.

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

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

Business Process Flow for Integration of Retail Merchandise Procure to Pay

This flowchart illustrates the process flow for Retail Merchandise Procure to Pay integration:



Business process flow for Retail Merchandise Procure to Pay

The overall process includes:

Work Location	Step
Oracle Retail (ReIM)	1. Configure and generate outgoing data from ReIM.
Integration process	2. The integration process checks for data in the ReIM Invoice Header and Invoice Lines interface tables. 3. Populate the Invoice cross-reference with the Retail key and common key. 4. Load and transform the Oracle Retail data into the PeopleSoft Voucher Header and Voucher Lines Staging tables.

Work Location	Step
	5. Delete the data from the source tables (ReIM Invoice Header and Invoice Lines interface tables) after a successful load.
PeopleSoft Payables	6. Run the Voucher Build process and import the vouchers into PeopleSoft Payables system for the pay cycle. 7. Invoke a process to populate the new PeopleSoft Invoice XREF table with the PSFT key and RETL key.

Prerequisites

Before performing this process integration, ensure that:

1. Suppliers and their locations, payment terms, and currency exchange rates are synchronized between the Oracle Retail and PeopleSoft applications.
2. Chart of accounts segments are mapped between the Oracle Retail and PeopleSoft applications.
3. Chart of accounts code combinations are validated.
4. Calendar must be setup for transactions.

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 mapping (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 already 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 an ORG_UNIT column to map it to BUSINESS_UNIT (AP).
9. Oracle Retail modifies the IM_AP_STAGE_HEAD and IM_AP_STAGE_DETAIL tables and adds segment 11 –segment 20 for the dynamic mapping to PeopleSoft ChartFields.
10. 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 decimal and Oracle Retail allows 4 digits. So some data may be rounded while moving from Oracle Retail to PeopleSoft.

Performing Setup Tasks

This section discusses:

- Setup tasks specific to ReIM
- Setup tasks specific to PeopleSoft Payables
- Setup tasks specific to the integration layer

Setup Tasks Specific to ReIM

For more information, see the *ReIM Operations Guide*, “PeopleSoft Enterprise Financials Integration” for details about the setup tasks specific to ReIM.

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 Chapter 8, “Setting Up the Participating Applications”, Setting Up PeopleSoft Applications

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 the AIAConfigurationProperties.XML file with these parameters:

```
<ModuleConfiguration
moduleName="RetailToPeopleSoftInvoiceIntegration">
  <Property name="FromMailAddress">john.doe@oracle.com</Property>
  <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 of time 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>
```

Configuring and Generating Data in Oracle Retail

This section discusses how to:

- Configure the data in Oracle Retail
- Generate the data in Oracle Retail

Configuring the Data in Oracle Retail

Before running any transaction data, these tasks must be done in ReIM:

- General ledger (GL) account cross-reference is setup.
- GL options are defined.

For more information, see the *ReIM User Guide*, “System Administration,” General Ledger Accounts

Generating the Data in Oracle Retail

In 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 discrepancies for documents have been resolved by resolution documents. Once all of the resolution documents for a matched invoice are built, and all of the RCA/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.

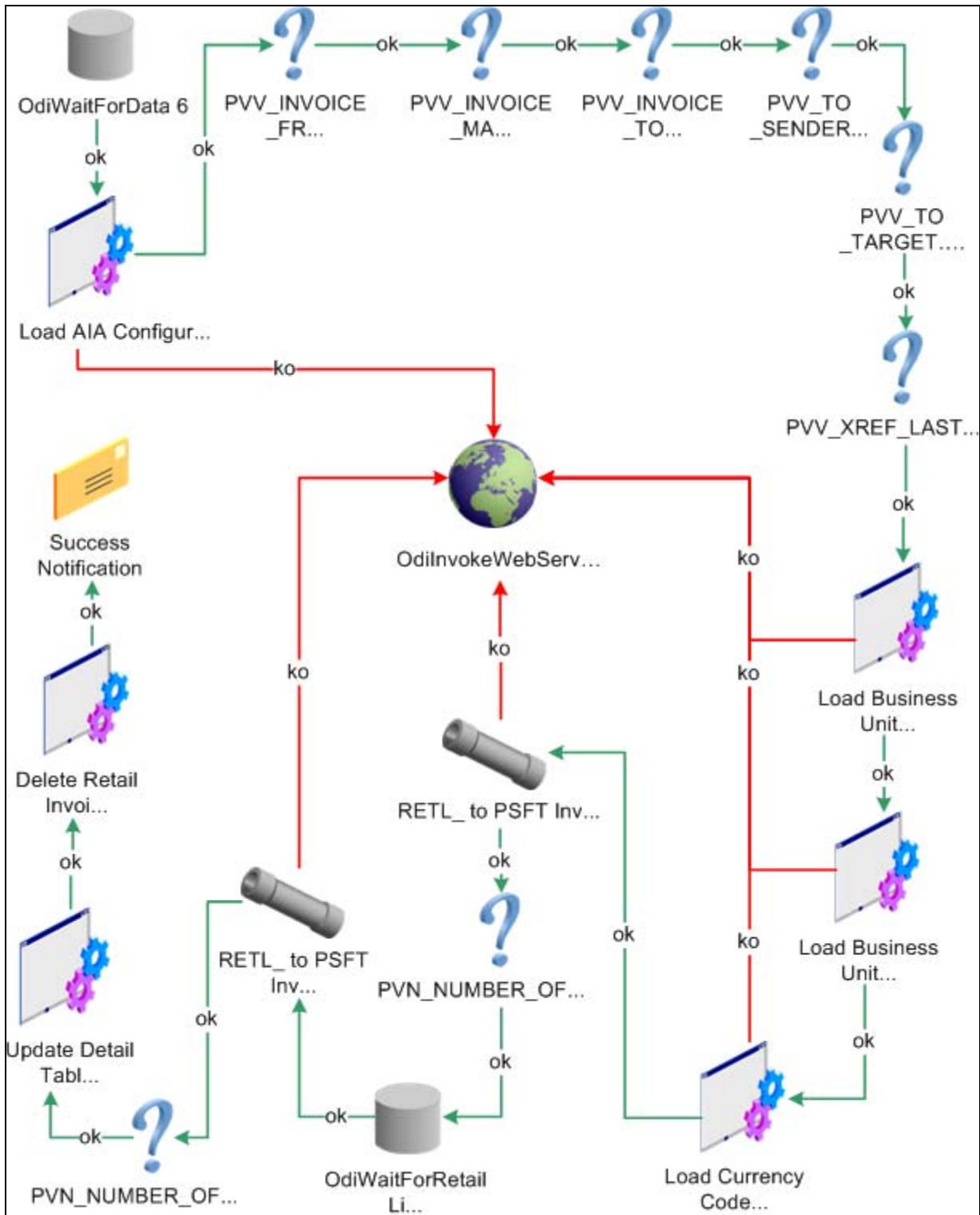
Once all of the transactions have been written, the process switches the status of the current invoices or documents to “Posted”, and then moves on to the next invoice or document. If a segment look-up fails, the failed record is written to a financials error table.

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

Picking Up and Transforming the Data

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

This flowchart illustrates the process:



Retail Merchandise Procure to Pay process flow

These tasks are performed 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 has been 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 ReIM.
4. ODI loads the BUSINESS_UNIT.xml and CURRENCY_CODE.XML DVM files into ReIM for mappings.
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 AIAAsyncErrorHandlingBPELProcess and the data is rolled back in the Retail tables, saving it for another run after the error is fixed.

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

This section discusses how to:

- Configure the process integration for Retail Merchandise Procure to Pay
- Set up a schedule
- Run the process integration for accounting entries

Configuring the Process Integration for Retail Merchandise Procure to Pay

After you have installed the process integration for Retail Merchandise Procure to Pay, you need to configure several parameters. Open the AIAConfigurationProperties.XML file that is available in the <AIA_HOME>/config folder.

Note: Whenever the AIAConfigurationProperties.XML file is updated, the file must be reloaded for updates to be reflected in the applications or services that use the updated properties. You can perform this reload by clicking the Reload button on the Configuration page in the Oracle AIA Console. Alternatively, you can perform the reload by rebooting the server.

For more information, see *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, “Using the BSR,” Loading Oracle AIA Configuration File Updates

Prerequisite

Verify that Process Integration Pack (PIP) ODI Artifacts (code) has been imported and connections are configured.

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 sender's email address for email notification.

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

For the MailServerName property, configure the valid mail server which will be used to send email notifications.

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

If desired, update the PollingInterval properties:

```
<Property name="PollingInterval">60000</Property>
```

For this process integration, 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>
```

Setting Up and Exporting DVMs

Set up these DVMs on the ESB console:

- BUSINESS_UNIT
- CHARTOFACCOUNTS_GLELEMENT
- CURRENCY_CODE

During the installation process, the Oracle Universal Installer prompts for a location to export the DVMs. Export these DVMs from ESB to that location.

For more information about how to create DVMs, see *Oracle Enterprise Service Bus Developer's Guide*, "Creating and Populating Domain-Value Maps"

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.

Creation of the views in the Oracle Retail database includes these steps:

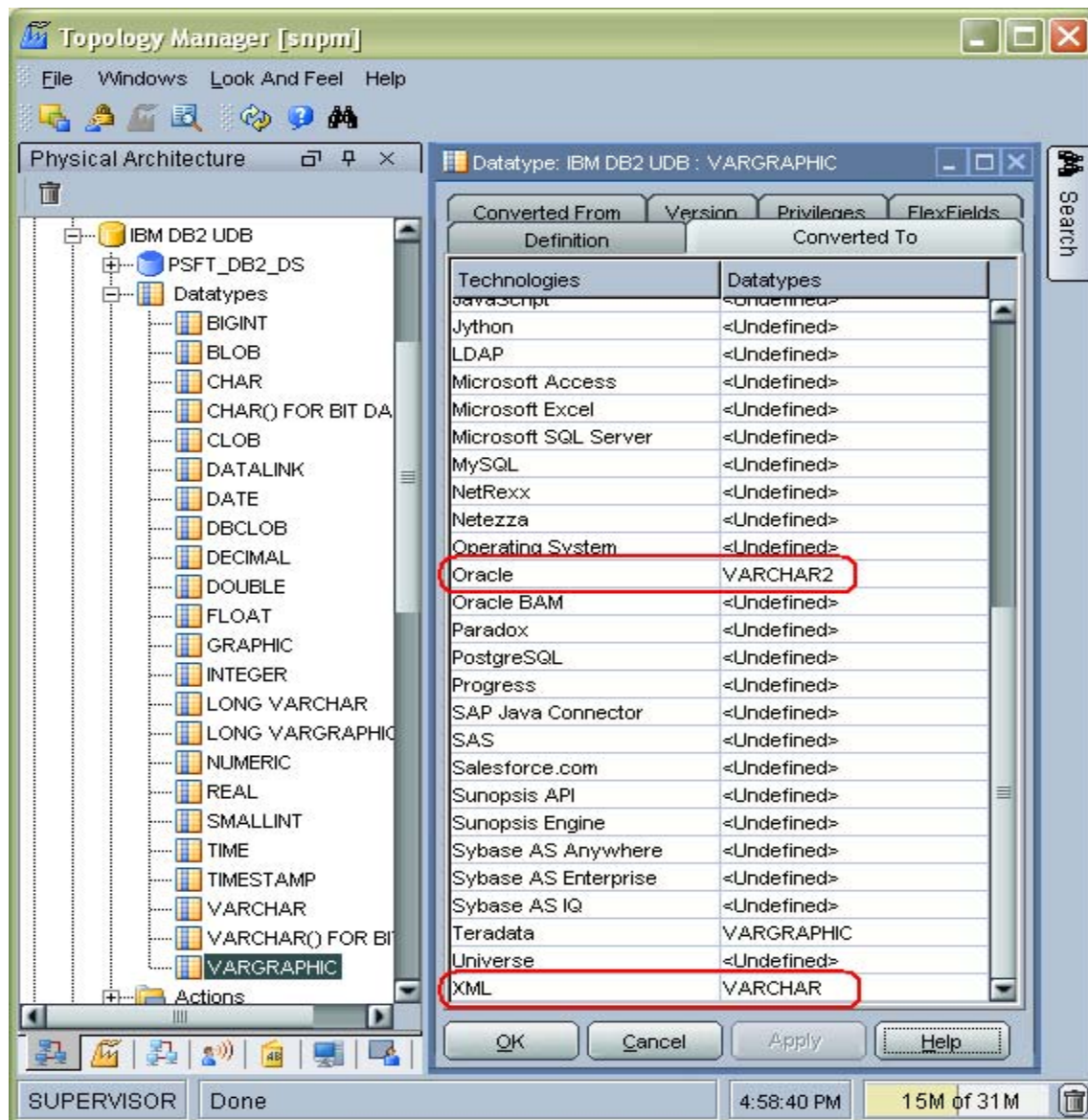
1. **Set up the cross-reference between segments and ChartFields.**
 - a. Update the DVM: In the ESB server, update the DVM CHARTOFACCOUNTS_GLELEMENT.
Map the Oracle Retail segments to the corresponding PeopleSoft ChartFields.
 - b. Export CHARTOFACCOUNTS_GLELEMENT DVM into the
<AIA_HOME>\PIPS\Core\DIS\RetailToPSFTFin\CreateRetailViewSQL folder.
2. **Run the Java program to generate the SQL scripts.**
 - a. From the command line, navigate to the CreateRetailViewSQL folder.
 - b. Ensure that the Java version is JDK 1.6.
 - c. Call the ANT script CreateRetailViewSQL.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 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

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 this image, before generating the scenario:



Set up the Datatype conversion to VARGRAPHIC

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

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?>\oracled\doc subdirectory.

Running the Process Integration for Retail Merchandise Procure to Pay

This process runs based on the schedules created for it. The integration between Oracle Retail and PeopleSoft Payables can be scheduled to occur automatically.

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

Please be aware that a standalone agent must not be configured with a datasource for master or workrepositories as this causes errors. The datasource is only supported on a J2EE agent.

Make sure you follow the configuration requirements for your selected agent and use according to the available Oracle Data Integration User Guide.

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.

Chapter 6: 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
- Service solution components
- Data requirements

Drill Back and Drill Forward Overview

Drill back and drill forward helps the PeopleSoft and Oracle Retail applications' users to access information about the financial transactions from the integrated systems in a seamless and intuitive manner. It provides the users end-to-end traceability between PeopleSoft Payables and General Ledger back to Oracle Retail merchandising System (RMS), Oracle Retail Sales Audit (ReSA), and Oracle Retail Invoice Matching (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 the PeopleSoft and Oracle Retail applications' users to trace and investigate the financial transactions by:

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

In this integration, the Application Integration Architecture (AIA) layer maintains the cross-references to associate the object identifications among the RMS, ReSA, and 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 ReIM to PeopleSoft Payables. The AIA services use the cross-references to link the invoices to payments.

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

- **RMS Stock Ledger:** This application is the source of transactions for the valuation of sellable inventory located in stores, warehouses, and distribution centers.
- **ReSA:** This application is the source of transactions for recognition of revenue and cash posting.
- **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 (RMS, ReSA, and 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.

Setup Activities

Modify the AIAConfigurationProperties.XML file with these parameters:

```
<ServiceConfiguration serviceName="DrillBackForward">
  <Property name="PSFTUrl.ProtocolHostPort"></Property>
  <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 Oracle Retail web service location respectively.

For more information, see Foundation Pack Install Guide.

Prerequisites

Before performing this integration, ensure that:

1. PeopleSoft GL and Payables and Oracle Retail implementations are live with all setup and configurations.
2. Accounting entries and invoice data have been integrated from Oracle Retail to the PeopleSoft applications.
3. BI Publisher has been installed and configured for Oracle Retail.
4. Invoice Oracle Data Integrator (ODI) flow is run.

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 XREF table is populated with data necessary to perform look up.

Drill Back and Drill Forward Details

This section discusses:

- Drill back details
- Drill forward details
- Overall Flow

Drill Back Details

The drill back enables the user to:

- Query 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 RMS and 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 RMS stock ledger from the PeopleSoft GL UI to find the source transactions for inventory valuation postings.

- Query ReIM from the PeopleSoft GL UI to find the source transactions for aged receipt write-offs, manually paid invoices, and matched pre-paid invoices.

This integration uses the DrillBackForwardURLService.

Drill Forward Details

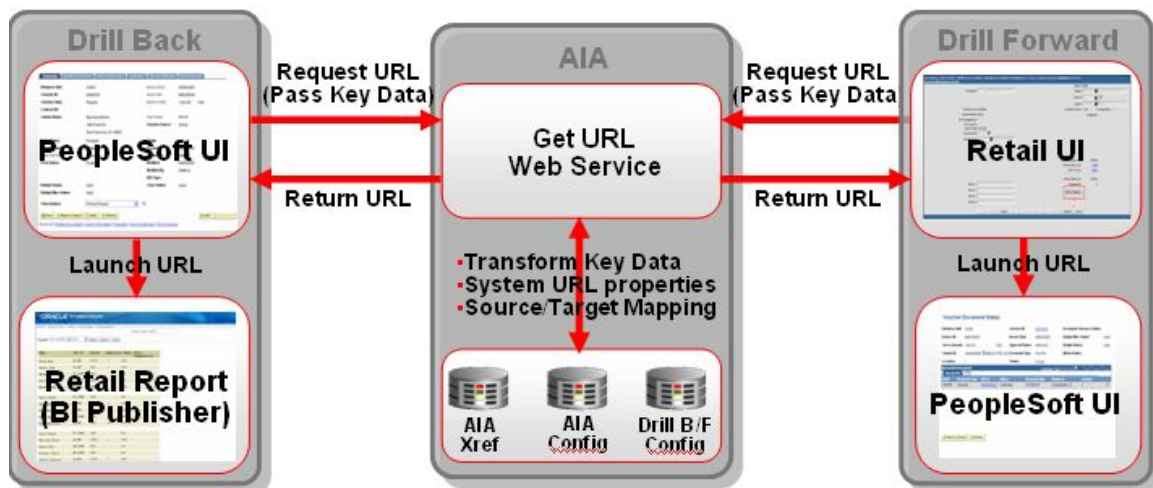
The drill forward enables the user to:

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

This integration uses the DrillBackForwardURLService.

Overall Flow

This diagram illustrates the overall drill back and drill forward flows:



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 requestor (RETAIL or PSFT).

3. The web service determines the target application URL based on the input parameter name and the web service requestor name.
4. The XREF 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.

Service Solution Components

This table contains the service solution components that are used in the process integration:

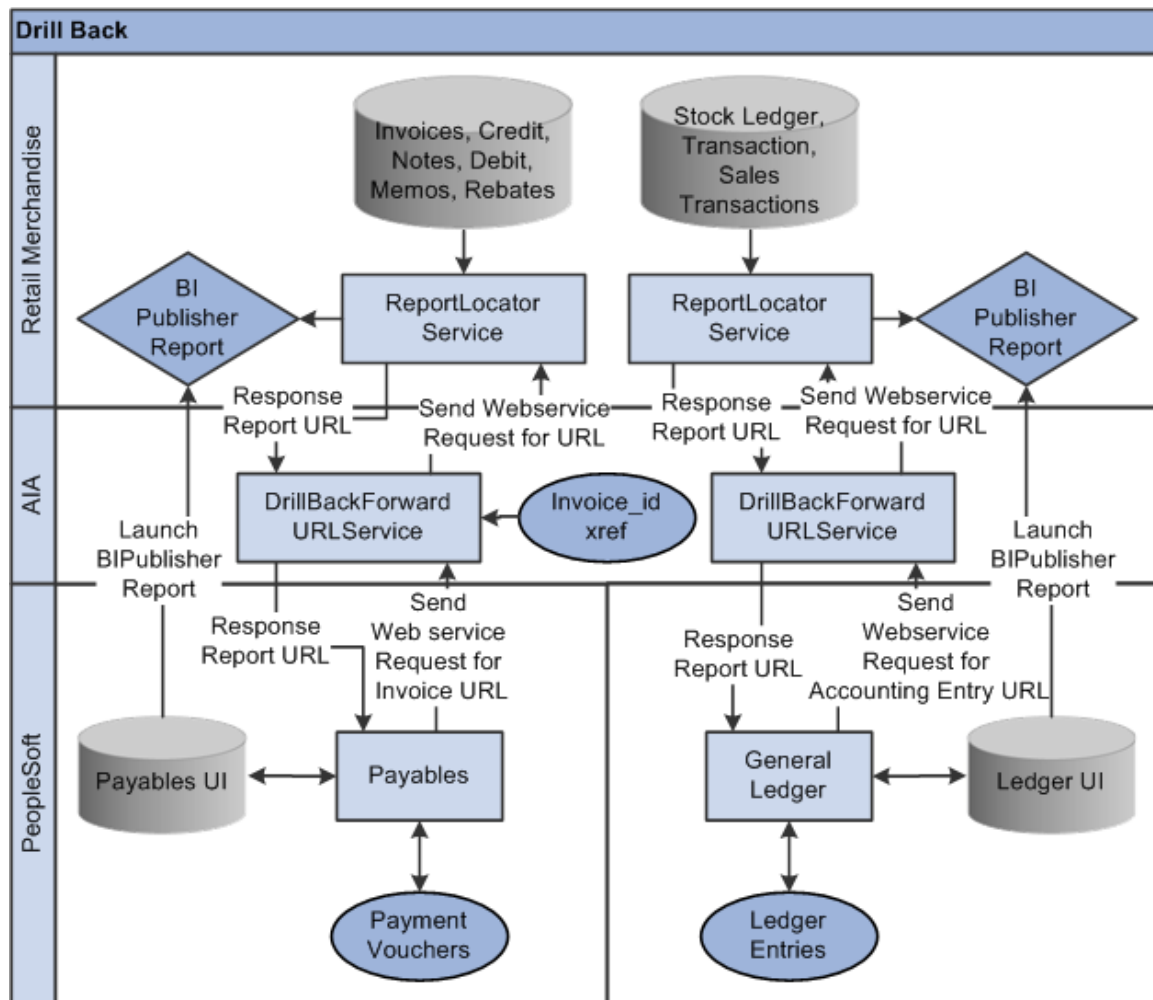
Name	Location	Caller	Object
DrillBackForwardURLService	AIA	Oracle Retail or PeopleSoft applications	PAYABLEINVOICE_PAYABLEINVOICEID or ACCOUNTING_ENTRY
Invoice Document Report	Retail BI Publisher	PeopleSoft Payables UI	BI Publisher URL
Aged Receipts Transaction Report	Retail BI Publisher	PeopleSoft GL 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

Service Solution Components 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.

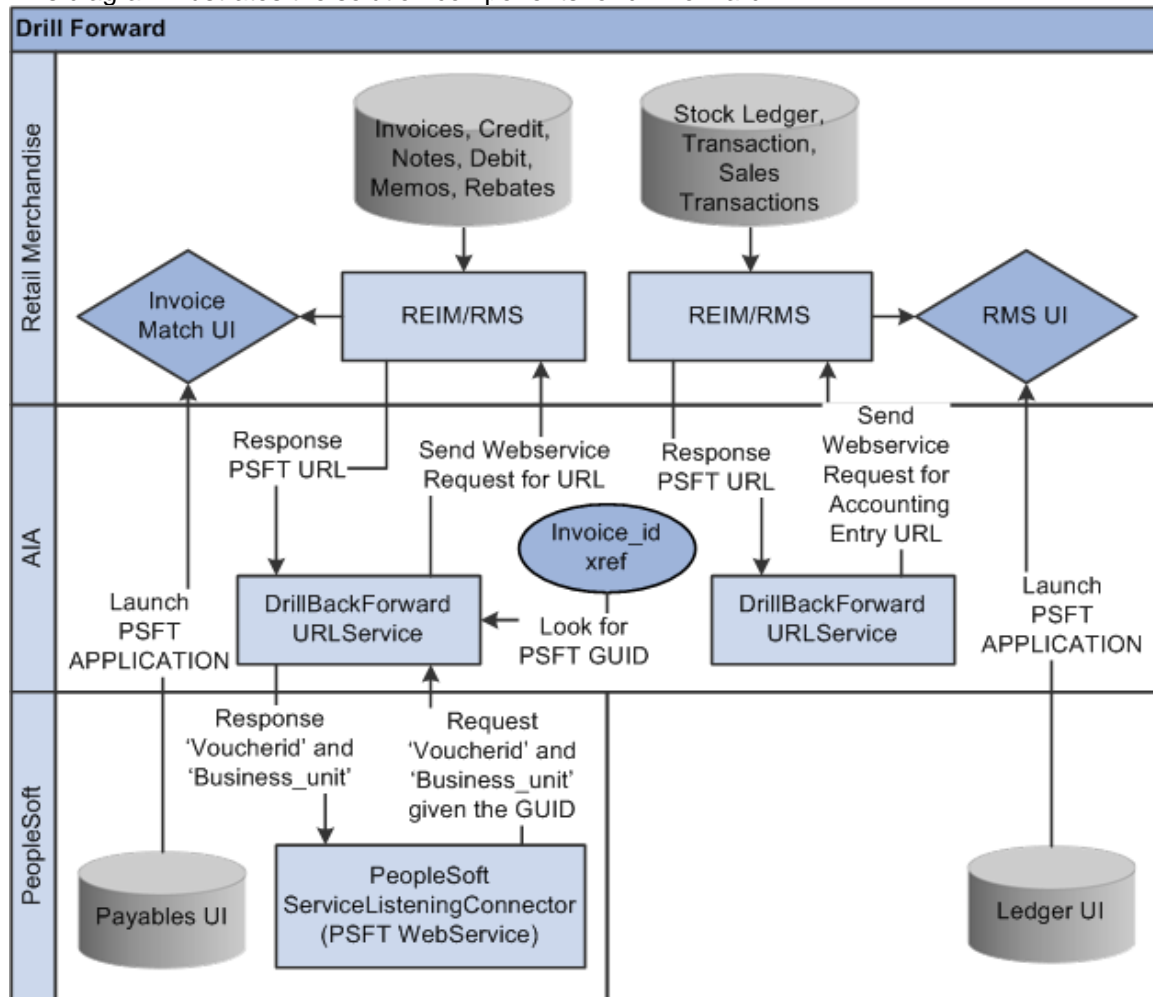
- **Invoice Document Report:** PeopleSoft Payables invokes the DrillBackForwardURLService for requesting a report on the invoices, credit notes, debit memos, and rebates from ReIM that made up the payment voucher. The AIA layer passes the URL to PeopleSoft Payables UI. PeopleSoft Payables UI initiates a report on the BIP server using the URL created by AIA. 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 ReIM that made up the journal entry. The AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the BIP server using the URL created by AIA. 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 ReIM that made up the journal entry. The AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the BIP server using the URL created by AIA. 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 ReSA that made up the journal entry. The AIA layer passes the URL to PeopleSoft GL. PeopleSoft GL UI initiates a report on the BIP server using the URL created by AIA. 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 RMS stock ledger that made up the journal entry. The AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the BIP server using the URL created by AIA. 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 RMS stock ledger that made up the journal entry. The AIA layer passes the URL to PeopleSoft GL UI. PeopleSoft GL UI initiates a report on the BIP server using the URL created by AIA. BIP generates a report for the requesting user.

This diagram illustrates the solution components for drill back:



Solution components for drill back

This diagram illustrates the solution components for drill forward:



Solution components for drill forward

Data Requirements

The accounting entry and invoice ODI integration flows are run.

For more information, see [Chapter 4: Process Integration for Accounting Entries](#) and [Chapter 5: Process Integration for Retail Merchandise Procure to Pay](#).

Chapter 7: Implementing Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials

This chapter discusses how to:

- Set up Oracle Retail applications
- Set up PeopleSoft General Ledger
- Set up PeopleSoft Payables
- Set up PeopleSoft Domain Value Maps
- Manually synchronize freight terms
- Manually synchronize currency rate types
- Manually synchronize currency codes
- Manually synchronize chart of accounts values
- Manually synchronize financial calendar
- Map business and organization units
- Work with cross-references
- Work with domain value maps (DVMs)
- Handle errors
- Set configuration properties

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

Setting Up Oracle Retail Applications

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

For more information, see the *ReIM Operations Guide*, “PeopleSoft Enterprise Financials Integration” and the *RMS Operations Guide, Volume 3*, “AIA Financials Integration” for setting up the Oracle Retail applications

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
- Setup recurring run controls for delay publish of effective dated objects
- Activate validation service operations
- Set up cross-reference tables
- Synchronize currency exchange rate

For more information, see PeopleSoft 9.0 Enterprise Installation Documentation Update: Oracle Retail Management Integration

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 delay publish of effective dated objects
- Activate validation service operations
- Set up cross-reference tables
- Synchronize payment terms and vendors

For more information, see PeopleSoft 9.0 Enterprise Installation Documentation Update: Oracle Retail Management Integration

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 need to load this table manually.

This section discusses:

- Populating cross-references

- Validating cross-references
- Viewing cross-reference data

Populating Cross-References

To populate the cross-references:

1. Create an xml file using the following sample.

Note: This script is only for initial load. It also enables customers to add new business unit required for their organizations. The process is same for both. It does not update the cross-reference table. There is no update capability.

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

Note: Substitute anything in bold and variables with the appropriate values. The sample uses only three row sets, create as many as needed. (Variables are preceded with a \$) The data for this file needs to be coordinated between PeopleSoft and Retail. No common column exists because this is auto generated by the tool.

Save the xml file to a temp directory on the SOA server in the bin directory (for your particular server).

\$SOA_HOME/integration/esb/bin

For example: \$SOA_HOME = /slot/ems1891/oracle/product/10.1.3.1/OracleAS_1

For more information about creating the cross-references, see *Oracle Enterprise Service Bus Developer's Guide 10g (10.1.3.4.0)*, Creating Cross References

2. Prior to run the xrefimport utility, set up the environment variables for the following:

- a. OC4J_USERNAME
- b. OC4J_PASSWORD

3. Run the xrefimport utility to import the cross-reference data using the steps 4 to 9.

4. Use Xterm or any other protocol to connect to the SOA server.

5. On the command line, enter the following commands:

Substitute your server and port. All commands are in bold.

For example:

Log in: **ssh -l \$USER_NAME \$HOST_NAME** (ssh -l is the lower case letter l, not numeric 1)

You might be asked the following questions:

"Are you sure you want to continue connecting (yes/no)?"
select **"yes"**

At the prompt for password, enter **\$LOGIN_PASSWORD**

You will get this prompt: -bash-3.00\$ (3.00 might be different)

6. At the bash prompt, change directories to the directory when the "bin folder" resides under your Oracle home:

For example:

-bash-3.00\$ cd \$OracleHome/integration/esb/bin

You will get the -bash prompt again.

7. Enter the following commands to import the data that is in the .xml file for the xref table.

This uses the xrefimport tool. After each command, you will be returned to the -bash prompt

- a. -bash-3.00\$ chmod +x xrefimport.sh
- b. -bash-3.00\$./xrefimport.sh -shell

(This is an optional command. It brings up the environment variables that need to be set)

- c. DB_USER
- d. DB_PASSWORD
- e. OC4J_USERNAME
- f. OC4J_PASSWORD

g. DB_URL

8. Set the environment variables mentioned in the step 7 with the correct values using the following script:

Example:

```
-bash-3.00$ export DB_USER=$db_user
```

```
-bash-3.00$ export DB_PASSWORD=$db_password
```

```
-bash-3.00$ export OC4J_USERNAME=$oc4j_username
```

```
-bash-3.00$ export OC4J_PASSWORD=$oc4j_password
```

```
-bash-3.00$ export DB_URL="jdbc:oracle:thin:@$HOST_NAME: $JDBC_PORT:$SID"
```

9. Import the data using the following command:

```
-bash-3.00$ ./xrefimport.sh -file \TEMP/$filename.xml -generate COMMON
```

Parameters:

-file = your xml file name with the directory name where you saved the xml file (your file should be in a TEMP folder in the bin directory).

-generate COMMON= this tells the tool to generate a common record for each set of rows with an auto-generated guid. It links the records together by row number (optional)

10. To view the data in the xref table in a document, use the export utility while in the command line. This utility exports the data from the xref table and creates a document with the specified name and location. If you have access to the database, you can copy the data into an excel spreadsheet.

For more information, see [Viewing Cross-Reference Data](#).

Validating Cross-References

To validate the 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 the following query:

```
select * FROM xref_data where xref_table_name='ORGANIZATION_ID'
```

This query retrieves all records in the ORGANIZATION_ID TABLE.

or

```
select *from xref_data where xref_table_name='ORGANIZATION_ID' and
```

```
xref_column_name='PSFT_01'
```

This query can be modified to change the column name to RETA_01 and COMMON.

Or

```
select * from xref_Data
where row_number in (select row_number from xref_Data
where value = 'CAN01')
```

The 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.

The data from SQL can be exported or copied to an excel spreadsheet. You can sort this data by row number, and so on.

For more information about cross-references, see [Working with Cross-References](#).

Viewing Cross-Reference Data

To view the cross-reference data

The data in the cross-reference table needs to be synched with the Business Unit DVM in PeopleSoft. Because the COMMON is an auto generated guid, and not known to the PeopleSoft customer, data visibility is essential. This data need to be entered in the PeopleSoft DVM.

1. **Exporting To File:** After the records have been written to the cross-reference table, the data can be viewed easily by exporting it to another document.

Run the xrefexport utility to export the cross-reference data using these instructions:

- a. Use Xterm or any other protocol to connect to the SOA Server.
- b. On the command line, type the following commands. Substitute with your server and port. Commands are bold.

For example:

Log in: `ssh -l $USER_NAME $HOST_NAME`

(ssh -l is the lower case letter l, not numeric 1)

You might be asked the following questions:

“Are you sure you want to continue connecting (yes/no)?” say “yes”

At the prompt for password, enter \$LOGIN_PASSWORD

You will get the following prompt:

-bash-3.00\$ (3.00 might be different)

- c. At the bash prompt, change directories to the directory when the “bin folder” resides under your Oracle home:

For example:

-bash-3.00\$ cd \$OracleHome/integration/esb/bin

You will get the -bash prompt again

- d. Enter the following commands to import the data that is in the xml file for the xref table. This will be using the xrefimport tool.

After each command, you will be returned to the -bash prompt

-bash-3.00\$ chmod +x xrefexport.sh

-bash-3.00\$./xrefexport.sh -shell (this command is optional, it will bring up the environment variables that needs to be set)

DB_USER

DB_PASSWORD

OC4J_USERNAME

OC4J_PASSWORD

DB_URL

- e. Set the mentioned environment variables with the correct values, using this script:

For example:

-bash-3.00\$ export DB_USER=\$db_user

-bash-3.00\$ export DB_PASSWORD=\$db_password

-bash-3.00\$ export OC4J_USERNAME=\$oc4j_username

-bash-3.00\$ export OC4J_PASSWORD=\$oc4j_password

-bash-3.00\$ export DB_URL="jdbc:oracle:thin:@\$HOST_NAME:\$JDBC_PORT:\$SID"

- f. Export the data using this command:

-bash-3.00\$./xrefexport.sh -file \TEMP/\$filename.xml -table ORGANIZATION_ID

Parameters:

-file = the tool will create a file for you with the name you give it. (This file can also have a .txt extension)

-table = this tells the tool which xref table to use for the export

2. Copy and paste from the SQL Worksheet:

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

- a. Query on the table
- b. `select * FROM xref_data where xref_table_name='ORGANIZATION_ID'`
- c. right click on the displayed data, select all/copy and paste into an excel spreadsheet

Setting Up PeopleSoft Domain Value Maps

Set up these PeopleSoft domain value maps (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 9.0 Enterprise Installation Documentation Update: Oracle Retail Management Integration

Manual Synchronization of 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 records. RMS needs the freight terms for purchase order generation.

PeopleSoft payables freight terms codes and their descriptions are manually set up in RMS. Incremental changes in freight terms are manually synchronized from PeopleSoft Payables to RMS. These are not synchronized automatically.

For more information, see the *PeopleSoft Enterprise Application Fundamentals 9.0 PeopleBook*, “Defining Financials and Supply Chain Management Common Definitions”

Manual Synchronization of 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, one exchange rate is selected. The selected currency exchange rate is used by all of the Oracle Retail products that integrate with PeopleSoft.

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 Domain Value Mapping (DVM) table.

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

Because the volume of currency exchange rate types is low, they are maintained and synchronized manually in PeopleSoft GL and RMS. These are not synchronized automatically. This synchronization is required for the synchronization of currency exchange rates between PeopleSoft GL and RMS.

For more information, see the *PeopleSoft Enterprise Global Options and Reports 9.0 PeopleBook*, “Processing Multiple Currencies”

Manual Synchronization of 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 9.0 PeopleBook*, "Working With Currencies and Market Rates"

Manual Synchronization of Chart of Accounts Values

A chart of accounts is a listing of all the accounts in the 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 General Ledger (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.

Chart of accounts is manually synchronized from PeopleSoft Payables to Oracle Retail. It is not synchronized 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, manual setup of the valid chart of accounts segment or ChartFields values for each business unit.
- Incremental chart of accounts synchronization: Manual synchronization of 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. The most current active Chart of Account value row needs to be manually synchronized from PeopleSoft GL to Oracle Retail.

The PeopleSoft system can also create future dated chart of accounts. These future dated charts of accounts need to be updated manually when they become effective.

After the chart of accounts values have been setup in both systems, a validation occurs for each Oracle Retail transaction ready to be transmitted to PeopleSoft system.

For more information, see the *PeopleSoft Enterprise Application Fundamentals 9.0 PeopleBook*, "Defining and Using ChartFields"

Manual Synchronization of Financial Calendar

The calendars are manually maintained in Oracle Retail and PeopleSoft; they are not automatically synchronized.

For more information, see the *PeopleSoft Enterprise Application Fundamentals 9.0 PeopleBook*, "Defining Accounting Calendars"

Mapping of Business and Organization Units

A *business unit* (BU) is an identification code that represents a high-level organization of business information. It is used 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 the business unit. 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 Application Integration Architecture (AIA) DVM table.

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

The 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 9.0 PeopleBook*, "Setting Up PeopleSoft Payables Business Units" and the *PeopleSoft Enterprise General Ledger 9.0 PeopleBook*, "Defining Your Operational Structure"

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 Application Integration Architecture - Foundation Pack - Integration Developer's Guide* and the *Oracle Cross Reference User Guide*.

This table lists the 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:To CurrencyCode:ConversionType: ConversionDate	
	PSFT_01	PeopleSoft GUID	
PAYABLEINVOICE_PAYABLEINV OICEID	COMMON	Same GUID as PSFT_01	Populated by the Oracle Data Integrator (ODI) invoice integration and

Cross Reference Table Name	Column Name	Description	Usage
			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	

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

Working with DVMs

DVMs are a standard feature of the Oracle Service Orientation 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 AIA layer. The 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.

This table lists the DVMs for the Oracle Retail to PeopleSoft Financials PIP:

DVM Type	DVM Column Name	Comments
BUSINESS_UNIT	COMMON, RETL_01, PSFT_01	This maps Oracle Retail's Set Of Books ID to PeopleSoft GL business unit.
CHARTOFACCOUNTS_ACCOUNTSTATUS	COMMON, RETL_01, PSFT_01	This is a hard coded DVM and should not be changed. This maps the PeopleSoft and EBO value of "true" or "false" to Oracle Retail's value of "valid" or "invalid".
CHARTOFACCOUNTS_GLELEMENT	COMMON, RETL_01, PSFT_01	This maps Oracle Retail's segment values to PeopleSoft GL elements.

DVM Type	DVM Column Name	Comments
ADDRESS_COUNTRYID	COMMON, RETL_01, PSFT_01	This maps the country codes between Oracle Retail and PeopleSoft system.
CURRENCY_CODE	COMMON, RETL_01, PSFT_01	This maps the currency codes between Oracle Retail and PeopleSoft system.
CURRENCYEXCHANGE_CONVERSIONTYPE CODE	COMMON, RETL_01, PSFT_01	This maps the currency exchange codes between Oracle Retail and PeopleSoft system.
LANGUAGE_CODE	COMMON, RETL_01, PSFT_01	This maps the language code between Oracle Retail and PeopleSoft system. This DVM is used to determine which language to be sent the translatable fields to Oracle Retail.
ORACLERETAILSEGMENTCOLNAMES	CHARTOFACCOUNTS_GLELEMENT, STG_FIF_GL_DATA, IM_FINANCIALS_STG, IM_AP_STAGE_DETAIL	This is a hard coded DVM and should not be changed. This lists the column names for all 20 segments for the tables in the RMS, REIM, and REIM Journal Entry systems. This is used for the accounting entry ODI flow.
STATE	COMMON, RETL_01, PSFT_01	This maps the state codes between Oracle Retail and PeopleSoft Payables.
SUPPLIERPARTY_ADDRESSTYPE	COMMON, RETL_01, PSFT_01	This maps the supplier's address type between Oracle Retail and PeopleSoft Payables.
SUPPLIERPARTY_STATUSCODE	COMMON, RETL_01, PSFT_01	This maps the supplier's status between Oracle Retail and PeopleSoft Payables.

Handling Errors

Based on the roles defined for the services, the system sends email notifications if a service errors out. This table lists the errors generated by the Oracle Retail to PeopleSoft Financials PIP services:

Error Code	Message Text
AIA_ERR_AIARTL2PSFT_1001	Invalid Requesting System; Requesting Systems must be RESA, REIM, or 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

The Integration Broker process can be set 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 AIA error handling, see the *Oracle Application Integration Architecture - Foundation Pack: Core Infrastructure Components Guide*, "Setting Up and Using Error Handling and Logging"

Viewing EBO Implementation Maps (EIMs)

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

Setting Configuration Properties

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

Note: Whenever the AIAConfigurationProperties.XML file is updated, the file must be reloaded for updates to be reflected in the applications or services that use the updated properties. You can perform this reload by clicking the Reload button on the Configuration page in the Oracle AIA Console. Alternatively, you can perform the reload by rebooting the server.

For more information, see the *Oracle Application Integration Architecture Core Components Guide*, "Using the BSR," Loading Oracle AIA Configuration File Updates.

Setting for the ProcessGLAccountValidationPeopleSoftProvABCSImpl service property:

Property Name	Value/Default Values	Description
---------------	----------------------	-------------

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code, defined in Business Service Repository (BSR), from the account validation provider.
Routing.GL_CHARTFIELD.1. PSFT_01.EndpointURI	http://\${psftfscm.http.host}:\${psftfscm.http.port}/PSIGW/PeopleSoftServiceList eningConnector/\${psftfscm.http.default.local.node}	PeopleSoft GL_Chartfield web service endpoint location. This is a SOAP endpoint URL.
Routing.GL_CHARTFIELD.1. CAVS.EndpointURI	http://\${http.hostname}:\${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.1. 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.

Setting for the SyncCurrencyExchangeListPeopleSoftReqABCSExtImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) 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.SyncCurrencyExchangeListPeopleSoftReqABCSExtImpl	http://\${http.hostname}:\${http.port}/AIAValidationSystemServlet/asy	CAVS Endpoint URI, when CAVS is enabled.

Property Name	Value/Default Values	Description
pl.CAVS.EndpointURI	ncrequestrecipient	
Routing.SyncCurrencyExchangeListPeopleSoftReqABCSImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the SyncPaymentTermListPeopleSoftReqABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) 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.SyncPaymentTermListPeopleSoftReqABCSImpl.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIAValidationSystemServlet/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.SyncPaymentTermListPeopleSoftReqABCSImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the CreatePaymentTermPeopleSoftReqABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) 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.

Property Name	Value/Default Values	Description
ABCSExtension.PostProcessABM	true/false Default = false	Controls whether to route messages to the extensibility system.
Routing.CreatePaymentTermPeopleSoftReqABCImpl.CAVS.EndpointURI	http://\$(http.hostname):\$(http.port)/AIAValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.CreatePaymentTermPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the UpdatePaymentTermPeopleSoftReqABCImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) 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.UpdatePaymentTermPeopleSoftReqABCImpl.CAVS.EndpointURI	http://\$(http.hostname):\$(http.port)/AIAValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.UpdatePaymentTermPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the CreateSupplierPartyPeopleSoftReqABCImpl service property:

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

Property Name	Value/Default Values	Description
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.hostname):\$(http.port)/AIValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.CreateSupplierPartyPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the UpdateSupplierPartyPeopleSoftReqABCImpl service property:

Property Name	Value/Default Values	Description
Sender.SystemID	PSFT_01	PeopleSoft system instance code (defined in BSR) 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.UpdateSupplierPartyPeopleSoftReqABCImpl.CAVS.EndpointURI	http://\$(http.hostname):\$(http.port)/AIValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.UpdateSupplierPartyPeopleSoftReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the SyncCurrencyExchangeListRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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.hostname}:\${http.port}/orabpel/default/SyncCurrencyExchangeListRetailProvJMSProducer/1.0	Oracle Retail currency exchange endpoint wsdl location. This is a URL that normally resides on the SOA server.
Routing.SyncCurrencyExchangeListRetailProvJMSProducer.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIAValidationSystemService/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.

Setting for the SyncPaymentTermListRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.PayTermService.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIAValidationSystemService/syncresponsesimulator	CAVS endpoint location to simulate a sync response for the Oracle Retail PayTermService endpoint wsdl location.
Retail.Language	1	The 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.

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.

Setting for the UpdatePaymentTermRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.PayTermService.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIValidationSystemService/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.

Setting for the CreatePaymentTermRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.PayTermService.CAVS	http://\${http.hostname}:\${http.port}/AIValidationSystemService	CAVS endpoint location to simulate a sync response for the Oracle Retail PayTermService

Property Name	Value/Default Values	Description
S.EndpointURI	t/syncresponsesimulator	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.

Setting for the SyncSupplierPartyListRetailProvABCImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.SupplierServicePort.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIValidationSystemService/t/syncresponsesimulator.	CAVS endpoint location to simulate a sync response for the Oracle Retail SupplierParty service endpoint wsdl location.
Retail.Language	1	The 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 not, then once we have determined if that supplier is active or inactive, we'll pass the corresponding value to Retail.
Retail.InctiveCode	I	Used to determine If the inbound supplier is active or not, then once we have determined if

Property Name	Value/Default Values	Description
		that supplier is active or inactive, we'll pass the corresponding value to Retail.

Setting for the CreateSupplierPartyRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.SupplierServicePort.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIAValidationSystemService/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.

Setting for the SyncSupplierPartyListPeopleSoftReqABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	PSFT_01	Oracle Retail system instance code (defined in BSR) from the supplier party provider.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages to the verification system.
Routing.SyncSupplierPartyListPeopleSoftReqABCSImpl.CAVS.EndpointURI	http://\${http.hostname}:\${http.port}/AIAValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.SyncSupplierPartyListPeopleSoftReqABCSImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Property Name	Value/Default Values	Description
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.

Setting for the UpdateSupplierPartyRetailProvABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) 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. This is a SOAP endpoint URL.
Routing.SupplierServicePort.CAVS.EndpointURI	http://{http.hostname}:{http.port}/AIValidationSystemService/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.

Setting for the ProcessGLAccountValidationRetailReqABCSImpl service property:

Property Name	Value/Default Values	Description
Default.SystemID	RETL_01	Oracle Retail system instance code (defined in BSR) from which messages originate.
Routing.RouteToCAVS	true/false Default = false	Controls whether to route messages

Property Name	Value/Default Values	Description
		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.hostname}:\${http.port}/AIValidationSystemService/asyncrequestrecipient	CAVS Endpoint URI, when CAVS is enabled.
Routing.ProcessGLAccountValidationRetailReqABCImpl.MessageProcessingInstruction.EnvironmentCode	PRODUCTION	This is a code that reflects what environment the ABCS is running in. This could be "PRODUCTION" or "CAVS".

Setting for the 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}	This value is the name of the PeopleSoft FSCM Application servlet.
PSFTUrl.Site	\${psftfscm.application.url.site}	This value is the name of the PeopleSoft FSCM Application site.
PSFTUrl.PortalNodeContent	EMPLOYEE/ERP/c	This value is the Port, Node, and Content for the PeopleSoft FSCM Application. The format of this information in this: <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. This is a SOAP endpoint URL.
RETAIL.URL.WS.Location	http://\${retail.rsl.host}:\${retail.rsl.port}/ReportLocatorBean/ReportLocatorService	Oracle Retail Report Locator service endpoint location. This is a SOAP endpoint URL.

Index

- Chart of Accounts Combination Validation
 - core components, 81
 - data requirements, 81
 - integration services, 81
 - Oracle Retail interfaces, 80
 - overview, 77
 - PeopleSoft GL, 80
 - solution assumptions and constraints, 78
- Chart of Accounts Combination Validation Integration Flow, 78
- ChartOfAccountsEBS, 82
- CreatePaymentTermPeopleSoftReqABCImpl, 50
- CreatePaymentTermRetailProvABCImpl, 53
- CreateSupplierPartyPeopleSoftReqABCImpl, 70
- CreateSupplierPartyRetailProvABCImpl, 74
- Currency Codes, 124
- Currency Exchange Rate Integration
 - core components, 27
 - data requirements, 27
 - details, 24
 - integration services, 27
 - overview, 21
 - PeopleSoft GL interfaces, 26
 - prerequisites, 22
 - RIB interfaces, 26
 - solution assumptions and constraints, 22
- Currency Exchange Rate Types, 124
- CurrencyExchangeEBS, 33
- CurrencyExchangePeopleSoftJMSConsumer, 29
- CurrencyExchangePeopleSoftJMSProducer, 28
- Domain Value Maps, 128
- Drill Back and Drill Forward
 - data requirements, 115
 - details, 110
 - overall flow, 111
 - overview, 108
 - prerequisites, 110
 - service solution components, 112
 - service solution components details, 112
 - setup activities, 109
 - solution assumptions and constraints, 110
- Error handling, 130
- Financial Calendar, 125
- Freight Terms, 124
- Implementing Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials
 - describing domain value maps, 128
 - handling errors, 130
 - identifying cross-references, 126
 - manual synchronization of chart of accounts values, 125
 - manual synchronization of currency codes, 124
 - manual synchronization of currency exchange rate types, 124
 - manual synchronization of financial calendar, 125
 - manual synchronization of freight terms, 124
 - mapping of business and organization units, 126
 - overview, 116

- setting configuration properties, 130
- setting up Oracle Retail applications, 116
- setting up PeopleSoft DVMs, 123
- setting up PeopleSoft GL, 117
- setting up PeopleSoft Payables, 117
- Integration of Accounting Entries
 - configuring data in Oracle Retail, 87
 - configuring the process integration, 90
 - facts and constraints, 86
 - generating data in Oracle Retail, 87
 - overview, 84
 - performing setup tasks, 86
 - picking up and transforming data, 88
 - prerequisites, 86
 - running the process integration, 94
 - setting up a schedule, 92
- Integration of Retail Merchandise Procure to Pay
 - configuring data in Oracle Retail, 100
 - configuring the process integration, 104
 - generating data in Oracle Retail, 100
 - overview, 95
 - performing setup tasks, 98
 - picking up and transforming data, 101
 - prerequisites, 97
 - running the process integration, 107
 - setting up a schedule, 105
 - solutions assumptions and constraints, 97
- Key Benefits of PIP, 10
- Life Cycle Management for Reference Data, 21
- Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials Overview, 7
- Overview
 - chart of accounts combination validation, 77
 - currency exchange rate integration, 21
 - drill back and drill forward, 108
 - integration of accounting entries, 84
 - integration of Retail Merchandise Procure to Pay, 95
 - Oracle Retail Merchandising Integration Pack for PeopleSoft Enterprise Financials, 7
 - payment term integration, 37
 - supplier integration, 56
- Participating Applications Overview, 10
 - PeopleSoft GL, 12
 - PeopleSoft Payables, 11
 - ReIM, 11
 - ReSA, 11
 - RMS, 10
- Payment Term Integration
 - core components, 46
 - data requirements, 43
 - details, 39
 - integration services, 47
 - multiple language support, 42
 - overview, 37
 - PeopleSoft Payables interfaces, 42
 - prerequisites, 37
 - RMS interfaces, 43
 - solution assumptions and constraints, 38
- PaymentTermEBS, 52
- PaymentTermPeopleSoftJMSConsumer, 48
- PaymentTermPeopleSoftJMSProducer, 48
- PayTermService, 55
- PeopleSoft DVMs, 123
- Process Integration for Reference Data, 21
 - currency exchange rate integration, 21

- payment term integration, 37
 - supplier information integration, 56
- ProcessGLAccountValidationRetailReqABCSImpl, 82
- Retail Merchandise Procure to Pay Business Process Flow, 16
- Sequence Diagram
 - currency exchange rate integration, 25
 - payment term integration, 40
 - supplier integration, 60
- Solution Assumptions and Constraints
 - chart of accounts combination validation, 78
 - currency exchange rate integration, 22
 - drill back and drill forward, 110
 - integration of Retail Merchandise Procure to Pay, 97
 - payment term integration, 38
 - solution assumptions and constraints for the PIP, 19
 - supplier information integration, 57
- SupplierPartyEBS, 72
- SupplierPartyPeopleSoftJMSConsumer, 68
- SupplierPartyPeopleSoftJMSProducer, 68
- Suppliers Information Integration
 - core components, 67
 - data requirements, 63
 - details, 58
 - integration services, 67
 - multiple language support, 61
 - overview, 56
 - PeopleSoft Payables interfaces, 62
 - prerequisites, 57
 - RMS interfaces, 62
 - solution assumptions and constraints, 57
- SyncCurrencyExchangeListPeopleSoftReqABCSImpl, 31
- SyncCurrencyExchangeListRetailProvABCSImpl, 34
- SyncCurrencyExchangeListRetailProvJMSProducer, 35
- SyncPaymentTermListPeopleSoftReqABCSImpl, 49
- SyncPaymentTermListRetailProvABCSImpl, 52
- SyncSupplierPartyListPeopleSoftReqABCSImpl, 69
- SyncSupplierPartyListRetailProvABCSImpl, 73
- UpdatePaymentTermPeopleSoftReqABCSImpl, 51
- UpdatePaymentTermRetailProvABCSImpl, 54
- UpdateSupplierPartyPeopleSoftReqABCSImpl, 71
- UpdateSupplierPartyRetailProvABCSImpl, 75