PeopleTools 8.4: PeopleSoft Integration Broker

PeopleSoft®

PeopleTools 8.4: PeopleSoft Integration Broker

SKU Tr84IBR-B 0302

PeopleBooks Contributors: Teams from PeopleSoft Product Documentation and Development.

Copyright © 2002 PeopleSoft, Inc. All rights reserved.

Printed in the United States.

All material contained in this documentation is proprietary and confidential to PeopleSoft, Inc. ("PeopleSoft"), protected by copyright laws and subject to the nondisclosure provisions of the applicable PeopleSoft agreement. No part of this documentation may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, including, but not limited to, electronic, graphic, mechanical, photocopying, recording, or otherwise without the prior written permission of PeopleSoft.

This documentation is subject to change without notice, and PeopleSoft does not warrant that the material contained in this documentation is free of errors. Any errors found in this document should be reported to PeopleSoft in writing.

The copyrighted software that accompanies this document is licensed for use only in strict accordance with the applicable license agreement which should be read carefully as it governs the terms of use of the software and this document, including the disclosure thereof.

PeopleSoft, the PeopleSoft logo, PeopleTools, PS/nVision, PeopleCode, PeopleBooks, *PeopleTalk,* and Vantive are registered trademarks, and "People power the internet." and Pure Internet Architecture are trademarks of PeopleSoft, Inc. All other company and product names may be trademarks of their respective owners. The information contained herein is subject to change without notice.

Contents

PeopleSoft Integration Broker Preface
About This PeopleBook xiii
Before You Begin xiii
PeopleSoft Application Fundamentals xiv
Related Documentation xiv
Hard-copy Documentation xiv
PeopleBooks Standard Field Definitions xv
Typographical Conventions and Visual Cues xvi
Page and Panel Introductory Table xvii
Comments and Suggestions xviii

Chapter 1

Understanding Integration Broker
Overview of Integration Broker 1-1
Integration Gateway 1-1
Integration Engine 1-2
Integration Gateway Architecture 1-2
Connectors 1-3
Gateway Manager 1-4
Gateway Services 1-4
Integration Engine Architecture 1-6

Chapter 2

Understanding Integrations
Integration Scenario Overview 2-1
Integrations with PeopleSoft 8.4 Systems 2-2
Configuration Tasks 2-2
Integrations with PeopleSoft 8.4 Systems Using Remote Gateways 2-4
Configuration Tasks 2-5
Integrations with PeopleSoft 8.4 Systems Using Hubs 2-7
Hub Routing Types 2-8
Generic-Routing Hub Configuration 2-8
Sender-Specified Routing Hub Configuration 2-10
Integrations with Third-Party Systems 2-13

Integrations with Third-Party Systems Using Remote Gateways....2-15
Sending Messages to Third-Party Systems Using Remote Gateways....2-15
Sending Messages from Third-Party Systems to PeopleSoft 8.4 Systems....2-17
Integrations with PeopleSoft 8.1 Systems....2-20
High-Level Configuration Steps....2-21

Chapter 3

Creating and Implementing Integrations

Establishing the Messaging Environment....3-1
Producing Basic Integrations....3-1
Understanding the Basic Messaging Elements....3-2
Understanding the Messaging Process Flow....3-3
Producing Advanced Integrations....3-4
Enhancing Basic Development and Administration....3-4
Applying Transaction Modifiers....3-5
Invoking Transform Programs....3-6

Chapter 4

Defining Message Channels

Understanding Message Channels....4-1
Defining a New Message Channel....4-2
Assigning Messages to the Channel....4-3
Configuring the Message Channel....4-3
Receiving a Subset of Messages....4-6

Chapter 5

Defining Messages

Understanding Messages....5-1
Creating Message Definitions....5-2
Defining a New Message....5-2
Understanding the Message Definition....5-3
Defining a Message Version....5-4
Inserting a New Version....5-5
Renaming a Version....5-5
Setting the Default Version....5-6
Using Records In a Message Definition....5-6
Understanding Message Record Structure....5-6
Inserting a Child Record....5-7
Inserting a Sibling Record....5-7
Reorganizing the Record Structure....5-8

Configuring Message Properties 5-8
Understanding Nonrepudiation 5-10
Accessing Message PeopleCode 5-11
Understanding Message PeopleCode 5-11
Creating a Message Subscription 5-12
Modifying Subscription PeopleCode 5-14
Accessing and Modifying OnRequest PeopleCode 5-15
Generating a Test Message 5-16

Chapter 6

Sending and Receiving Messages

Understanding Sending and Receiving Messages 6-1
PeopleSoft Rowsets 6-1
XML Document Object Model 6-2
Simple Object Access Protocol 6-3
Transaction Process Flow 6-4
Understanding PSCAMA 6-5
The PSCAMA Record 6-6
Language Codes 6-7
Audit Action Codes 6-7
Other PSCAMA Considerations 6-8
Generating and Sending a Message 6-9
Understanding Outbound Messaging 6-9
Handling Outbound Asynchronous Transactions 6-10
Handling Outbound Synchronous Transactions 6-11
Handling Cookies 6-14
Receiving and Processing a Message 6-15
Handling Inbound Asynchronous Transactions 6-15
Handling Inbound Synchronous Transactions 6-22
Processing Inbound Errors 6-24
Validating Data 6-24
Using the Exit Function 6-25
Correcting Errors 6-27

Chapter 7

Administering Basic Integrations

Understanding Integration Administration 7-1
Configuring a Gateway 7-2
Understanding Gateway Configuration 7-2
Defining a New Gateway 7-3

Specifying the Gateway Location 7-4
Refreshing the Gateway Properties 7-4
Registering Installed Target Connectors 7-5
Editing Connector Properties 7-6
Configuring a Node 7-8
Understanding Node Configuration 7-8
Defining a New Node 7-8
Specifying General Node Information 7-9
Specifying Contact Information 7-13
Defining Node Properties 7-13
Specifying a Connector 7-14
Adding Connector Properties 7-16
Modifying Property Values 7-17
Configuring Transactions 7-17
Understanding Transactions 7-17
Viewing the Transaction List 7-18
Defining a Transaction 7-19
Editing Transaction Details 7-21
Editing Transaction Message Details 7-24
Editing Transaction Connector Properties 7-25

Chapter 8

Using Integration Broker Monitor

Understanding Asynchronous Messages 8-2
Brokers, Contractors and Queues 8-2
Message Server Processes 8-3
Dispatchers and Handlers 8-4
Asynchronous Publication Flow and Message Status 8-5
Asynchronous Subscription Flow and Message Status 8-8
Understanding Synchronous Messages 8-10
Synchronous Publication Flow and Message Status 8-10
Synchronous Subscription Flow and Message Status 8-11
Getting Started Using Integration Broker Monitor 8-12
Opening Integration Broker Monitor 8-12
Setting Security and Permissions 8-13
Understanding Message Status Information, Blocked Channels and Stalled Channels 8-13
Message Status Information 8-13
Blocked Channels 8-14
Stalled Channels 8-15

Monitoring Messaging System Information 8-15
Working with the Monitor Messages Component 8-15
Monitoring System Queue Information 8-19
Monitoring Asynchronous Message Instances 8-20
Monitoring Publication Contracts 8-20
Monitoring Subscription Contracts 8-21
Monitoring Synchronous Message Instances 8-21
Monitoring Channel Status 8-21
Working with Node Status 8-22
Working with Pub/Sub Server Domains 8-24
Running Message Queries 8-27
Monitoring Asynchronous Message Details 8-28
Opening the Message Details Component 8-28
Viewing Asynchronous Message Properties 8-29
Viewing Asynchronous Message Errors 8-32
Viewing Asynchronous Messages in XML Format 8-32
Viewing Nonrepudiation Information for Asynchronous Messages 8-33
Viewing Synchronous Message Details 8-33
Opening the Synchronous Details Component 8-33
Viewing Synchronous Message Details 8-34
Viewing Synchronous Message Errors 8-35
Running Batch Error Notification Processes 8-36
Running Batch Message Archiving Processes 8-38
Purging Messaging Tables 8-39
Using the Message Monitor Component Interface 8-39

Chapter 9

Understanding Error Handling, Logging, Tracing and Debugging

Integration Gateway Error Handling 9-1
Target Connector Error Handling 9-1
Listening Connector Error Handling 9-2
Integration Gateway Exception Types 9-2
Standard Exceptions 9-2
Integration Gateway Message and Error Logging 9-4
Integration Gateway Message Logging 9-5
Integration Gateway Error Logging 9-7
Application Server Logging and Tracing 9-8
Application Engine Tracing 9-9
Debugging Subscription PeopleCode 9-9
Integration Broker Debugging Quick Reference 9-9

Chapter 10

Applying Transformation, Translation and Filtering

Understanding Transformation, Translation and Filtering 10-1
Third Party Considerations 10-2
Understanding the PeopleSoft Base Message Format 10-3
Timestamp Format 10-3
The FieldTypes Section 10-3
The MsgData Section 10-4
A Message Example 10-5
Developing Transform Programs 10-8
Defining a Transform Program 10-9
Working with Transform Programs 10-10
Accessing Message Data 10-11
Making Working Data Available Globally 10-13
Filtering Messages and Generating Errors 10-14
Working with a PeopleCode Filtering Example 10-15
Generating an Error 10-17
Applying Transformations 10-17
Using XSLT for Transformation 10-18
Performing Data Translation 10-19
Understanding Data Translation 10-20
Defining a Codeset Group 10-22
Defining a Codeset 10-23
Defining Codeset Values 10-25
Using XSLT for Data Translation 10-27
Working with an XSLT Translation Example 10-29

Chapter 11

Administering Relationships

Understanding Relationships 11-1
Transaction Modifiers 11-2
Determining Relationship Parameters 11-3
Selecting a Transaction Type Combination 11-3
Determining Where to Define the Relationship 11-4
Configuring a Relationship 11-5
Defining a New Relationship 11-5
Specifying a Node Pair 11-6
Overriding Node Properties 11-8
Overriding Contact Information 11-8
Managing Transaction Modifiers 11-9

Understanding the Trans Modifiers Page 11-9
Defining a Transaction Modifier 11-10
Editing Transaction Modifier Details 11-12
Applying Asynchronous to Synchronous Modifiers 11-15
Retaining Messages at a Hub Node 11-16

Chapter 12

Using the Integration Gateway

Setting Integration Gateway Properties 12-1
Setting Integration Gateway Version Properties 12-2
Setting Integration Gateway Class Installation Properties 12-2
Setting Delivered Connector Configuration Properties 12-2
Setting Logging Properties 12-6
Setting Proxy Web Server Properties 12-7
Setting Integration Gateway Certificates Properties 12-8
Setting JMS Configuration Properties 12-8
Understanding Message Compression and Encoding 12-10
Working with Target Connectors 12-10
Understanding Target Connectors 12-11
Understanding Target Connector Properties 12-12
Using the HTTP Target Connector 12-13
Using the PeopleSoft Target Connector 12-15
Using the PeopleSoft 8.1 Target Connector 12-15
Using the FTP Target Connector 12-16
Using the JMS Target Connector 12-17
Using the SMTP Target Connector 12-22
Using the POP3 Target Connector 12-23
Using the Simple File Target Connector 12-35
Working with Listening Connectors 12-35
Understanding Listening Connectors 12-36
Using the HTTP Listening Connector 12-37
Using the PeopleSoft Listening Connector 12-42
Using the PeopleSoft 8.1 Listening Connector 12-42
Using the JMS Listening Connector 12-43
Setting Up Gateway Certificates 12-45
Understanding Gateway Certificates 12-45
Setting Up Gateway Certificates 12-46
Running the Integration Gateway Behind a Proxy Server 12-48
Understanding Proxy Authentication 12-48
Setting Gateway-Level Connector Properties 12-48

Chapter 13

Using the Integration Broker Connector SDK

Understanding the Integration Broker Connector SDK 13-1
Integration Broker Connector SDK Contents 13-2
Integration Broker Connector SDK Location 13-2
Integration Broker Connector API Documentation 13-3
Understanding Connector Development and Implementation 13-3
Connector Development Infrastructure 13-4
General Connector Class Development Considerations 13-7
Target Connector Class Development Considerations 13-8
Listening Connector Class Development Considerations 13-13
Installing Connector Classes 13-18
Registering Connectors 13-18
Connector Templates 13-18
Using the Java XML DOM Wrapper for Connector Development 13-23
Working with the Java XML DOM Wrapper 13-23
Using the Java XML DOM Wrapper Classes 13-24
Java XML DOM Code Example 13-24
Understanding Developing and Implementing Connectors in the C/C++ Environment 13-26
Development Process 13-26
Creating Target Connectors for the C/C++ Environment 13-28
Reusing Connector Code 13-31
Reusing Connector Code through Inheritance 13-31
Reusing Connector Code through Delegation 13-32
Testing Message Processing Using Send Master 13-33
Understanding Send Master 13-33
Opening Send Master 13-34
Understanding Send Master Workspaces 13-35
Understanding Send Master Project Types 13-40
Understanding Headers Used in Send Master 13-41
Creating Input File Projects 13-42
Working with Integration Broker Projects 13-44
Creating Integration Broker Projects 13-49
Creating Groups of Projects 13-53
Managing Groups of Projects 13-53
Testing Groups of Projects 13-54
Viewing Test Output 13-54
Sharing Projects and Groups 13-55
Posting Third-Party Messages to the Integration Gateway using the Simple Post Tool 13-56

Getting Started Using the Simple Post Tool 13-56
Understanding the Simple Post Tool..... 13-57
Using the Simple Post Tool 13-58

Glossary

Index

PeopleSoft Integration Broker Preface

PeopleSoft Integration Broker is a middleware technology that facilitates synchronous and asynchronous messaging among internal systems and trading partners, while managing message structure, message format, and transport disparities.

The "About This PeopleBook" section contains general product line information, such as related documentation, common page elements, and typographical conventions. This book also contains a glossary with useful terms that are used in PeopleBooks.

See **PeopleSoft Glossary**.

About This PeopleBook

This book provides you with the information that you need for implementing and using *PeopleTools 8.4* applications. Complete documentation for this release is provided on the CD-ROM PT84PBR0.

Note. Your access to PeopleSoft PeopleBooks depends on which PeopleSoft applications you've licensed. You may not have access to all of the PeopleBooks.

This section contains information that you should know before you begin working with PeopleSoft products and documentation, including PeopleSoft-specific documentation conventions, information specific to each PeopleSoft product line, and information on ordering additional copies of our documentation.

Before You Begin

To benefit fully from the information covered in this book, you should have a basic understanding of how to use PeopleSoft applications. We recommend that you complete at least one PeopleSoft introductory training course.

You should be familiar with navigating the system and adding, updating, and deleting information by using PeopleSoft windows, menus, and pages. You should also be comfortable using the World Wide Web and the Microsoft® Windows or Windows NT graphical user interface.

Because we assume that you already know how to navigate the PeopleSoft system, much of the information in these books is not procedural. That is, these books do not typically provide step-by-step instructions on using tables, pages, and menus. Instead, we provide you with the information that you need to use the system most effectively and to implement your PeopleSoft application according to your organizational or departmental needs. PeopleBooks expand on the material covered in PeopleSoft training classes.

PeopleSoft Application Fundamentals

Each PeopleSoft application PeopleBook provides implementation and processing information for your PeopleSoft database. However, there is additional, essential information describing the setup and design of your database contained in a companion volume of documentation called *PeopleSoft Application Fundamentals.*

PeopleSoft Application Fundamentals contains important topics that apply to many or all PeopleSoft applications across each product line. Whether you are implementing only one PeopleSoft application, some combination of products within a product line, or an entire PeopleSoft system, you should be familiar with the contents of this central PeopleBook. It contains fundamental information such as setting up control tables and administering security.

The PeopleSoft Applications Fundamentals PeopleBook contains common information pertinent to all applications in each product line, such as defining general options. If you're upgrading from a previous PeopleSoft release, you may notice that we've removed some topics or topic headings from the individual application PeopleBooks and consolidated them in this single reference book. You'll now find only application-specific information in your individual application PeopleBooks. This makes the documentation as a whole less redundant. Throughout each PeopleBook, we provide cross-references to *PeopleSoft Application Fundamentals* and other PeopleBooks.

Related Documentation

You can order printed, bound versions of the complete PeopleSoft documentation delivered on your PeopleBooks CD-ROM and additional copies of the PeopleBooks CDs through the Documentation section of the PeopleSoft Customer Connection website: http://www.peoplesoft.com/corp/en/login.asp

You can find updates and additional documentation for this release, as well as previous releases, on PeopleSoft Customer Connection (http://www.peoplesoft.com/corp/en/login.asp). Through the Documentation section of Customer Connection, you can download files to add to your PeopleBook library. You'll find a variety of useful and timely materials, including updates to the full PeopleSoft documentation delivered on your PeopleBooks CD.

Important! Before you upgrade, it is *imperative* that you check PeopleSoft Customer Connection for updates to the upgrade instructions. We continually post updates as we refine the upgrade process.

Hard-copy Documentation

To order printed, bound volumes of the complete PeopleSoft documentation delivered on your PeopleBooks CD-ROM, visit the PeopleSoft Press website from the Documentation section of PeopleSoft Customer Connection. The PeopleSoft Press website is a joint venture between PeopleSoft and Consolidated Publications Incorporated (CPI), our book print vendor.

We make printed documentation available for each major release shortly after the software is shipped. Customers and partners can order printed PeopleSoft documentation by using any of the following methods:

Internet	From the main PeopleSoft Internet site, go to the Documentation section of Customer Connection. You can find order information under the Ordering PeopleBooks topic. Use a Customer Connection ID, credit card, or purchase order to place your order. PeopleSoft Internet site: http://www.peoplesoft.com/.
Telephone	Contact Consolidated Publishing Incorporated (CPI) at **800 888 3559.**
Email	Send email to CPI at **callcenter@conpub.com**.

PeopleBooks Standard Field Definitions

Throughout our product documentation, you will encounter fields and buttons that are used on many application pages or panels. This section lists the most common fields and buttons and provides standard definitions.

Field	*Definition*
As of Date	The last date for which a report or process includes data.
Business Unit	An identification code that represents a high-level organization of business information. You can use a business unit to define regional or departmental units within a larger organization.
Description	Freeflow text up to 30 characters.
Effective Date	Date on which a table row becomes effective; the date that an action begins. For example, if you want to close out a ledger on June 30, the effective date for the ledger closing would be July 1. This date also determines when you can view and change the information. Pages or panels and batch processes that use the information use the current row. For more information about effective dates, see **Understanding Effective Dates** in ***Using PeopleSoft Applications.***
EmplID (employee ID)	Unique identification code for an individual associated with your organization.
Language or **Language Code**	The language in which you want the field labels and report headings of your reports to print. The field values appear as you enter them. Language also refers to the language spoken by an employee, applicant, or non-employee.

Field	*Definition*
Process Frequency group box	Designates the appropriate frequency in the **Process Frequency** group box: **Once** executes the request the next time the batch process runs. After the batch process runs, the process frequency is automatically set to **Don't Run**. **Always** executes the request every time the batch process runs. **Don't Run** ignores the request when the batch process runs.
Report ID	The report identifier.
Report Manager	This button takes you to the Report List page, where you can view report content, check the status of a report, and see content detail messages (which show you a description of the report and the distribution list).
Process Monitor	This button takes you to the Process List page, where you can view the status of submitted process requests.
Run	This button takes you to the Process Scheduler request page, where you can specify the location where a process or job runs and the process output format.
	For more information about the Report List page, the Process List page, and the Process Scheduler, see **Process Scheduler Basics** in the PeopleTools documentation.
Request ID	A request identification that represents a set of selection criteria for a report or process.
User ID	The system identifier for the individual who generates a transaction.
SetID	An identification code that represents a set of control table information or TableSets. A TableSet is a group of tables (records) necessary to define your company's structure and processing options.
Short Description	Freeflow text up to 15 characters.

Typographical Conventions and Visual Cues

We use a number of standard conventions and visual cues in our online documentation.

The following list contains our typographical conventions and visual cues:

`(monospace font)` Indicates a PeopleCode program or other program example.

Bold Indicates field names and other page elements, such as buttons and group box labels, when these elements are

	documented below the page on which they appear. When we refer to these elements elsewhere in the documentation, we set them in Normal style (not in bold). We also use boldface when we refer to navigational paths, menu names, or process actions (such as **Save** and **Run**).
Italics	Indicates a PeopleSoft or other book-length publication. We also use italics for *emphasis* and to indicate specific field values. When we cite a field value under the page on which it appears, we use this style: ***field value.*** We also use italics when we refer to words as words or letters as letters, as in the following: Enter the number *0,* not the letter *O.*
KEY+KEY	Indicates a key combination action. For example, a plus sign (+) between keys means that you must hold down the first key while you press the second key. For ALT+W, hold down the ALT key while you press W.
Cross-references	The phrase **For more information** indicates where you can find additional documentation on the topic at hand. We include the navigational path to the referenced topic, separated by colons (:). Capitalized titles in *italics* indicate the title of a PeopleBook; capitalized titles in normal font refer to sections and specific topics within the PeopleBook. Here's an example: **For more information,** see **Documentation on CD-ROM** in *About These PeopleBooks:* Additional Resources.

Note. Text in this bar indicates information that you should pay particular attention to as you work with your PeopleSoft system. If the note is preceded by **Important!,** the note is crucial and includes information that concerns what you need to do for the system to function properly.

Text in this bar indicates cross-references to related or additional information.

Warning! Text within this bar indicates a crucial configuration consideration. Pay very close attention to these warning messages.

Page and Panel Introductory Table

In the documentation, each page or panel description in the application includes an introductory table with pertinent information about the page. Not all of the information will be available for all pages or panels.

Usage	Describes how you would use the page or process.
Object Name	Gives the system name of the panel or process as specified in the PeopleTools Application Designer. For example, the Object Name of the Detail Calendar panel is DETAIL_CALENDAR1.
Navigation	Provides the path for accessing the page or process.
Prerequisites	Specifies which objects must have been defined before you use the page or process.
Access Requirements	Specifies the keys and other information necessary to access the page. For example, **SetID** and **Calendar ID** are required to open the Detail Calendar page.

Comments and Suggestions

Your comments are important to us. We encourage you to tell us what you like, or what you would like to see changed about our documentation, PeopleBooks, and other PeopleSoft reference and training materials. Please send your suggestions to:

PeopleSoft Product Documentation Manager
PeopleSoft, Inc.
4460 Hacienda Drive
Pleasanton, CA 94588

Or send comments by email to the authors of the PeopleSoft documentation at:

DOC@PEOPLESOFT.COM

While we cannot guarantee to answer every email message, we will pay careful attention to your comments and suggestions. We are always improving our product communications for you.

CHAPTER 1

Understanding Integration Broker

Overview of Integration Broker

PeopleSoft Integration Broker is a middleware technology that facilitates synchronous and asynchronous messaging among internal systems and trading partners, while managing message structure, message format, and transport disparities. Because of its modular design, many of the elements you develop for an integration can be reused in other integrations.

Integration Broker is comprised of two high-level subsystems, the Integration Gateway and the Integration Engine. The Integration Gateway resides on a PeopleSoft web server, and the Integration Engine is installed on a application server as part of your PeopleSoft application.

Integration Gateway

The **Integration Gateway** is a platform that manages the actual receipt and delivery of messages passed among systems through the Integration Broker. It provides support for the leading TCP/IP protocols used in the marketplace today, and more importantly, provides extensible interfaces for the development of new connectors for communication with legacy, Enterprise Resource Planning (ERP), and Internet-based systems.

Additional features of the Integration Gateway include:

- Backward compatibility for Extensible Markup Language (XML) Links and Application Messaging.
- Listening connectors and target connectors that transport messages between integration participants and the Integration Engine. Also allows customers to build their own custom connectors to complement those delivered with Integration Broker.
- Basic logging information concerning message receipt, delivery, and errors.
- Connection persistence where there are continuous open feeds to external system via connectors, with full failover capabilities.
- Transport protocol and message format management so that when messages reach the Integration Engine they are in the PeopleSoft message format.

See Also

Integration Gateway Architecture

Integration Engine

The **Integration Engine** runs on the PeopleSoft application server. It's tied closely to your PeopleSoft application, and produces or consumes messages for the application. Rather than communicating directly with other applications, the Integration Engine sends and receives messages through one or more separately installed integration gateways.

Features of the Integration Engine include:

- Has a modular architecture, so it can treat gateways as black boxes, and communicate with them using their standard connectors.
- Can adapt elements of an existing integration to produce a new integration with only minor adjustments.
- Handles messages containing data in a variety of formats, including PeopleSoft rowsets, XML document object model, Simple Object Access Protocol, and unstructured data.
- Sends and receives messages asynchronously (like email) or synchronously (suspending activity to wait for a response).
- Applies message transmission type and routing based on specifications you define in a PIA component.
- Can transform message structure and translate data content according to specifications you define in PIA components and apply with Extensible Stylesheet Language Transformation (XSLT) code or PeopleCode. These specifications can be reused for other integrations.
- Can handle security features like authentication, nonrepudiation and cookies.

See Also

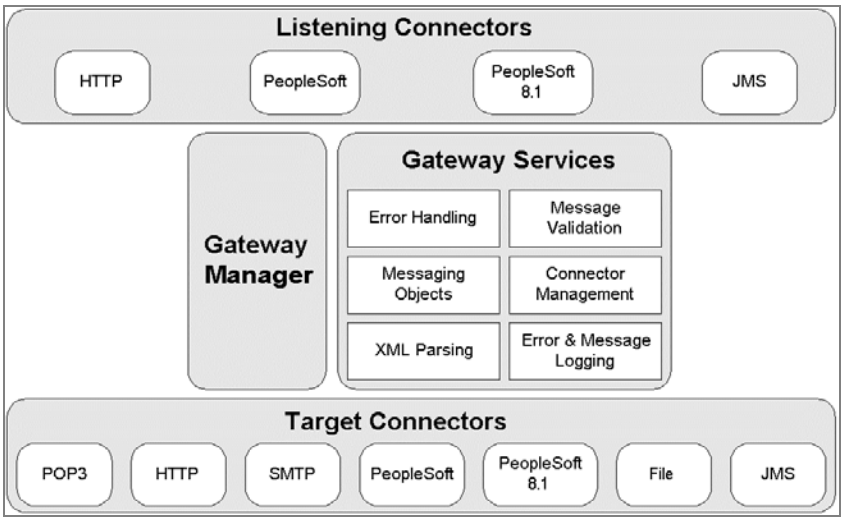
Integration Engine Architecture

"Creating and Implementing Integrations"

Integration Gateway Architecture

The Integration Gateway is primarily a conduit that receives and sends messages among integration participants' systems. The Integration Gateway is designed with flexibility and extensibility in mind. Messages are received through listening connectors and are delivered from target connectors. The Gateway Manager is a dispatcher for messages that flow though the Integration Gateway.

Listening connectors receive incoming messages and hand the incoming requests to the Gateway Manager. The primary responsibility of the Gateway Manager is to determine which target connector to use to properly deliver the messages to their intended recipients. The target connector then delivers the messages to the intended recipients using the their preferred protocol.



Integration Gateway Architecture

This section discusses:

- Target connectors and listening connectors.
- Integration Broker Connector Software Development Kit (SDK)
- Gateway Manager
- Gateway Services

Connectors

In the Integration Gateway architecture, listening connectors and target connectors transport messages between integration participants and the Integration Engine. These connectors support asynchronous, synchronous, and polling-based message handling. In addition, many of the connectors are configurable, based on user-defined settings at the Integration Gateway and node level.

Target Connectors

Target connectors open communication with other PeopleSoft systems or third-party systems and perform various operations. A target connector may or may not receive a response from the target system during each operation.

Not all inputs are driven by listening connectors. Some inputs are actually "pulled" into the Integration Broker using target connectors. An example of a delivered target connector that behaves in this manner is the POP3 target connector. With the POP3 target connector, an event, in this case a batch process is triggered in the Integration Engine and a request is generated to check email. The request arrives at the POP3 target connector, which interprets the request as instructions to check e-mail. New e-mail is downloaded from the POP3 server, and a response is generated. The response is returned to the Integration Engine and goes through standard *input* processing, such as authentication, authorization and routing.

Listening Connectors

Listening connectors receive incoming data streams and perform services based on the content of the stream. They are invoked externally by other systems, such as other PeopleSoft systems, third-party systems, and so forth.

Integration Broker Connector Software Development Kit (SDK)

The Integration Gateway provides a fully extensible model for developing new connectors built to the interface specification of the Integration Broker SDK by PeopleSoft customers, consultants, application developers, and so forth.

See Also

"Using the Integration Gateway," Working with Target Connectors

"Using the Integration Gateway," Working with Listening Connectors

"Using the Integration Broker Connector SDK"

Gateway Manager

The **Gateway Manager** processes every message that flows through the Integration Gateway and maintains links among the other major Integration Gateway components, including target connectors, listening connectors, and each of the Gateway Services.

Listening connectors invoke the Gateway Manager when they receive a message request. The Gateway Manager uses handles on the messaging objects, IBRequest and IBResponse, to determine how to route each message.

The Gateway Manager uses a number of the Gateway Services during this stage to perform operations such as message validation. The Gateway Manager then invokes the appropriate target connector based on the contents of the message object, and waits for a response back from the target connector. When the response is received, the Gateway Manager forwards the response back to the calling listening connector.

If an error occurs, the Gateway Manager may use the Error Handling service, and works with the service to prepare an error response for the listening connector.

See Also

Gateway Services

Gateway Services

This section describes the gatway services used by the Gateway Manager.

Error Handling

The Integration Gateway provides a standard error handling interface that is exposed to each connector. This service provides error handling and error logging for most connectors delivered with PeopleSoft Integration Broker.

Messaging Objects

Two objects represent the messaging objects service in the Integration Gateway:

- IBRequest
- IBResponse

These objects are central to the system as they represent the request and response that go in and out of the Integration Broker. See "Using the Integration Broker Connector SDK," Integration Broker Connector API Documentation.

XML Parsing

Most IBRequests and IBResponses that are processed in the system usually contain a Content section which represents the actual business message sent.

Most of the time these Content sections contain XML data. Since this is the case, often connectors must parse and traverse XML. Many developers find that the standard Java XML objects are cumbersome for manipulating XML, so the Integration Gateway provides an XML Parsing service that consists of objects that provide a very intuitive interface for manipulating XML objects. This service is delivered as a set of classes: XmlDocument, XmlNode and XmlNodeList. See "Using the Integration Broker Connector SDK," Integration Broker Connector API Documentation.

Message Validation

Messages that pass into the Integration Broker must contain certain elements in order for them to be processed. Since the Integration Gateway is the first component in the Integration Broker to process messages sent to a PeopleSoft application, the Integration Gateway performs basic message validation, such as making sure the message identifies its requestor and message name, to ensure that the Integration Engine and the target application can process them.

Connector Management

The Connector Management service is actually a composite of several smaller services the Integration Gateway implements to manage connectors. The Gateway processes each IBRequest to determine the appropriate connector to call in each situation. This is primarily a message routing function that has varying levels of complexity abstracted from the connectors. The connector manager also processes the IBResponse returned by each connector.

Error and Message Logging

The Integration Gateway provides a standard logging interface that most components in the system use, most notably the connectors.

Each PeopleSoft-delivered connector uses the logging API in the same fashion, ensuring that an administrator can quickly drill-down on problems or simply review message logs to see the IBRequest, IBResponses, and even the raw data sent to and received from integration participants.

See Also

Understanding Error Handling, Logging, Tracing and Debugging

Integration Engine Architecture

The Integration Engine uses a variety of PeopleTools elements to create, implement, manage and enhance integrations. Its modular architecture separates the integration development activity from the administrative activity.

The Integration Engine is really a combination of Application Designer definitions, PIA definitions, PeopleCode and XSLT code, along with the underlying mechanisms that tie all these elements together, including the *request handlers* that process both inbound and outbound messages according to the specifications in the development and administrative elements. See "Creating and Implementing Integrations" for a complete introduction to using these elements.

Elements of Basic Integration

- *Basic development elements:* Application Designer definitions and their associated PeopleCode. You can develop these elements independent of the messaging environment in which they'll be used. They are:
  - Message channel definition
  - Message definition
  - Sending PeopleCode
  - Subscription PeopleCode
  - OnRequest PeopleCode
- *Basic administrative elements:* PIA definitions that describe the messaging environment, including the participating applications, their local gateways, transmission types, and routing instructions. They are:
  - Gateway definition
  - Node definition
  - Transaction

Elements of Advanced Integration

- *Advanced development elements:* Application Engine programs and their associated code for filtering and modifying messages. They are:
  - Transform program
  - XSLT action
  - PeopleCode action
- *Codeset repository:* PIA definitions that specify how message data should be translated based on the source and target applications involved, the message definition, transmission type and direction, and other factors. The repository can be referenced by transform programs which apply the translations. It consists of:
  - Codeset group definition
  - Codeset definition
  - Codeset values definition
- *Advanced administrative elements:* PIA definitions that apply advanced routing options and transform programs. They are:
  - Relationship definition
  - Transaction modifier

Note. You can also apply advanced message routing using OnRouteSend and OnRouteReceive PeopleCode. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

See Also

"Creating and Implementing Integrations"

CHAPTER 2

Understanding Integrations

This chapter provides an overview of six basic integration scenarios you can implement using PeopleSoft Integration Broker.

This chapter discusses integrations with:

- PeopleSoft 8.4 systems.
- PeopleSoft 8.4 systems using remote gateways.
- PeopleSoft 8.4 systems using hub configurations.
- Third-party systems.
- Third-party systems using remote gateways.
- PeopleSoft 8.1 systems.
- High-level Integration Broker configuration steps.

Integration Scenario Overview

Prior to configuring PeopleSoft Integration Broker, you should install PeopleTools as well as PeopleSoft Internet Architecture (PIA). During the installation of these products, the Integration Engine and the Integration Gateway are automatically installed.

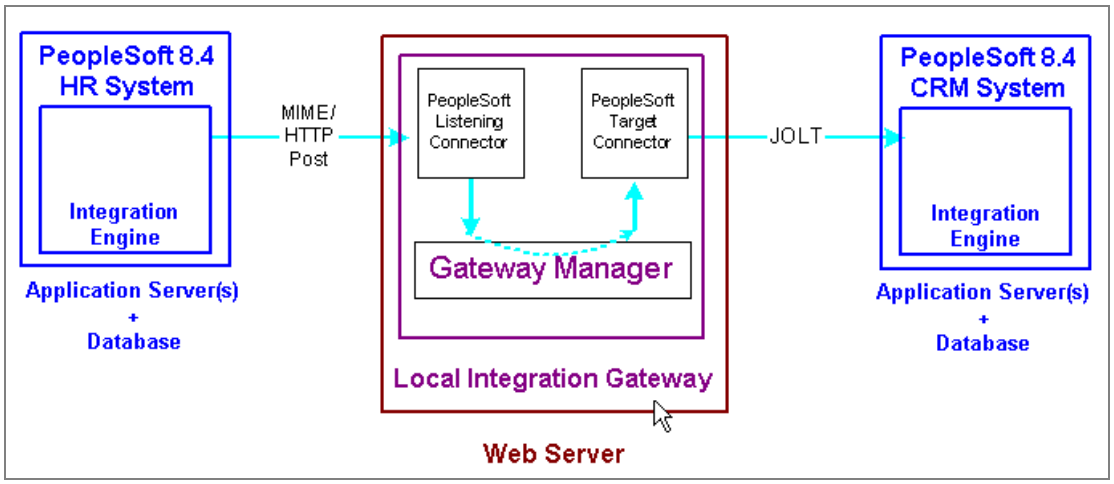
For each integration scenario discussed in this chapter, a configuration diagram using sample systems is provided that shows the integration components involved. In addition, an overview of the high-level configuration steps for each scenario is also provided. In general, the high-level tasks you will perform to configure any of the integration scenarios are:

- Define a local Integration Gateway.
- Define a remote Integration Gateway.
- Set up a local node.
- Set up a remote node.
- Set Integration Gateway properties.
- Set up inbound and outbound transactions.

You may not need to perform all these tasks. For instance, if you do not need to communicate over a firewall, you probably will not need to define and configure a remote gateway.

Note. As you review the information in this chapter keep in mind that Integration Broker employs definitional message routing by way of transactions that you define at the node level. PeopleSoft 8.1 Application Messaging employed content-based routing where all routing information was defined in the header of each message.

Integrations with PeopleSoft 8.4 Systems



Scenario: Integrations with PeopleSoft 8.4 Systems

The diagram shows a PeopleSoft 8.4 HR system communicating with a PeopleSoft 8.4 CRM system, and shows the configuration and interaction of Integration Broker components. This communication could be synchronous or asynchronous.

Using the systems shown in the graphic as an example, this section provides an overview of the configuration tasks required to implement this scenario.

Configuration Tasks

This section describes the source system and destination system configuration tasks, based on the scenario shown in the previous diagram.

PeopleSoft HR System Configuration Tasks

In this example, the PeopleSoft HR system is the source system. Perform the following tasks on that system:

- **Define a local Integration Gateway**. Use the Gateways component to define the local HR gateway.

- **Set up a local node**. The local node you set up represents the HR system. Use the Node Definition component to perform this task.
- **Set up a remote node**. The remote node you set up represents the CRM system. When you set up the remote node, specify the PeopleSoft Target Connector.
- **Set up outbound transactions**. Set up outbound transactions on the CRM (remote) node.

PeopleSoft CRM System Configuration Tasks

In this example, the PeopleSoft CRM system is the destination system. Perform the following tasks on that system:

- **Set up a local node**. The local node you set up represents the CRM system. Use the Node Definition component to perform this task.
- **Set up a remote node**. The remote node you set up represents the HR system. Use the Node Definition component to perform this task.
- **Set up inbound transactions**. Set up inbound transaction on the HR system (remote node).

Integration Gateway Configuration Tasks

The only required property you must set for the local gateway is the Jolt connect string(s) that enable communication to the destination PeopleSoft 8.4 CRM system. Use the IntegrationGateway.properties file to set this property.

See Also

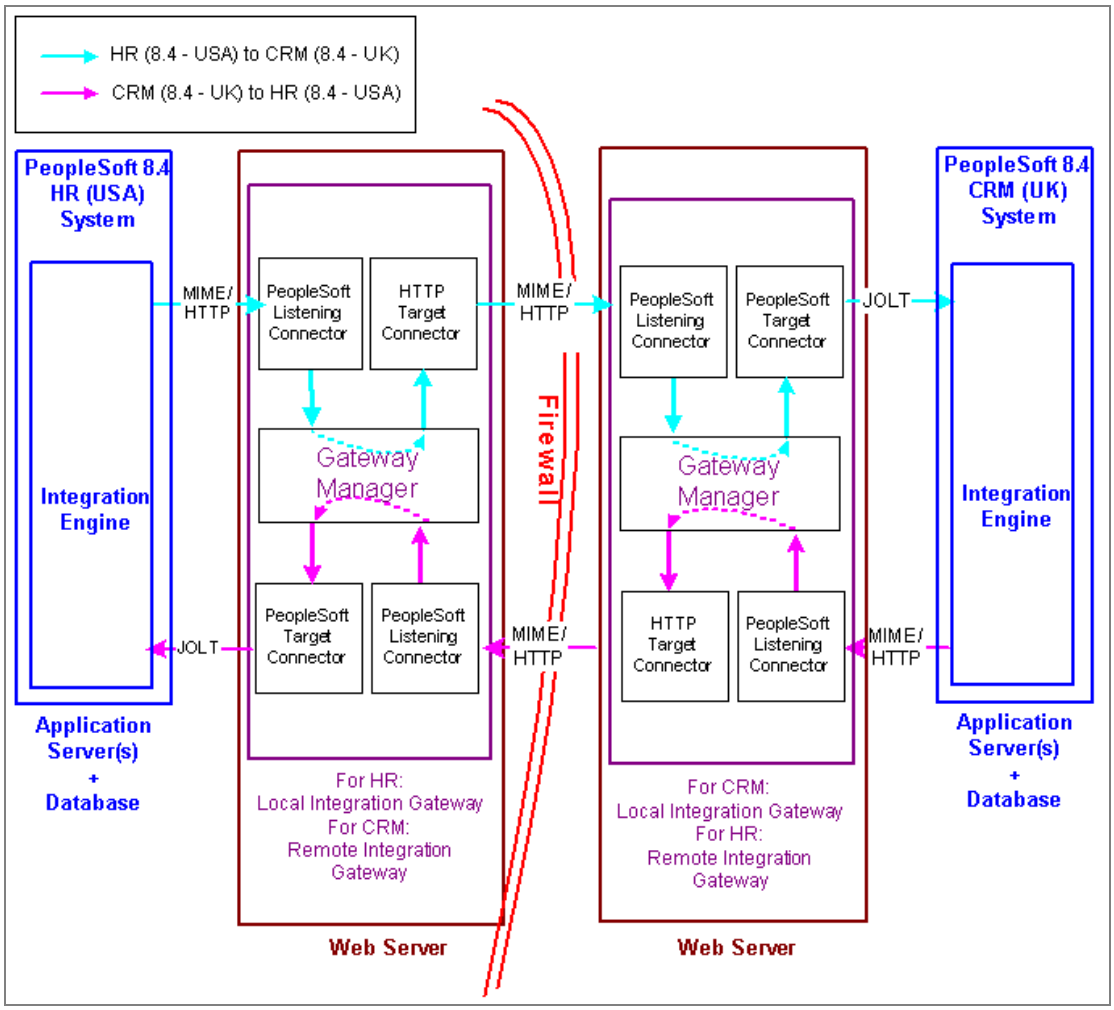
High-Level Configuration Steps

“Administering Basic Integrations“ Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

“Administering Basic Integrations” Configuring Transactions

Integrations with PeopleSoft 8.4 Systems Using Remote Gateways



Scenario: Integrations with PeopleSoft 8.4 systems using remote gateways.

You can use a remote gateway configuration for integrations with PeopleSoft 8.4 systems in instances where connections with an integration participant are not possible through the Internet. This type of implementation enables you to communicate with Wide Area Networks (WANs) and Local Area Networks (LANs) where a firewall is present.

The diagram shows the configuration of Integration Broker components for integrations between other PeopleSoft 8.4 systems using a remote gateway. For this configuration scenario one PeopleSoft 8.4 system and one Web server (where the local Integration Gateway is installed) reside on each side of the firewall. The Web server on which the Integration Gateway resides can be on the same physical machine on which you have installed the PeopleSoft 8.4 application, or it can reside on its own machine. Each of these integration participants will require some configuration.

In this configuration scenario the Integration Broker uses the default remote gateway connector, the HTTP Target Connector, on the local gateway to send messages to the PeopleSoft Listening Connector on the remote gateway. Routing all messages through the

local gateway allows the Integration Broker to keep a complete centralized log of all integration messages.

Configuration Tasks

Since this example shows two-way communication, the PeopleSoft HR (USA) system and the PeopleSoft CRM (UK) system are source systems when outbound messages originate from them, and they are destination systems when messages are sent to them.

This section describes the configuration tasks for each of the components shown in the previous diagram. Keep in mind the following as you review these configuration tasks:

- PeopleSoft recommends using a single Integration Gateway for all applications that reside on one side of a firewall.
- The local Integration Gateway for one application is the remote Integration Gateway for the other application.

PeopleSoft HR (USA) System Configuration Tasks

On the PeopleSoft HR (USA) system:

- **Define a local Integration Gateway**. Use the Gateways component to define the local HR (USA) gateway.
- **Define a remote Integration Gateway**. The remote Integration Gateway for the PeopleSoft HR (USA) system is the CRM (UK) gateway. Use the Gateways component to define a new gateway, and specify the URL of the CRM (UK) gateway.
- **Set up a local Node**. The local node you set up represents the HR (USA) system. Use the Node Definitions component to perform this task.
- **Set up a remote Node**. The remote node you set up represents the CRM (UK) system. When you set up the remote node specify the CRM (remote) Integration Gateway and the PeopleSoft Target Connector on that gateway.
- **Set up inbound transactions**. Set up inbound transactions on the HR (local) node. Use the Transaction tab in the Node Definitions component for this task.
- **Set up outbound transactions**. Set up outbound transactions on the CRM (remote) node. Use the Transaction tab in the Node Definitions component for this task.

PeopleSoft HR (USA) Integration Gateway

The only required Integration Gateway property you must set for the local Integration Gateway is the Jolt connect string(s) that enable communication with the Integration Engine on the PeopleSoft HR (USA) system. Use the IntegrationGateway.properties file to set this property.

PeopleSoft CRM (UK) System Configuration Tasks

On the PeopleSoft CRM (UK) system:

- **Define a local Integration Gateway**. Use the Gateways component to define the local CRM (UK) gateway.
- **Define a remote Integration Gateway**. The remote gateway for the PeopleSoft CRM (UK) system is the HR (USA) gateway. Use the Gateways component to define a new gateway, and specify the URL of the HR (USA) gateway.
- **Set up a local Node**. The local node you set up represents the CRM (UK) system. Use the Node Definitions component to perform this task.
- **Set up a remote Node**. The remote node you set up represents the HR (USA) system. When you set up the remote node specify the HR (remote) Integration Gateway and the PeopleSoft Target Connector on that gateway.
- **Set up inbound transactions**. Set up inbound transactions on the HR (remote) node.
- **Set up outbound transactions**. Set up outbound transactions on the HR (remote) node.

PeopleSoft CRM (UK) Integration Gateway

The only required property Integration Gateway property you must set for the local UK Integration Gateway is the Jolt connect string(s) that enable communication with the Integration Engine on the destination PeopleSoft CRM (UK) system. Use the IntegrationGateway.properties file to set this property.

See Also

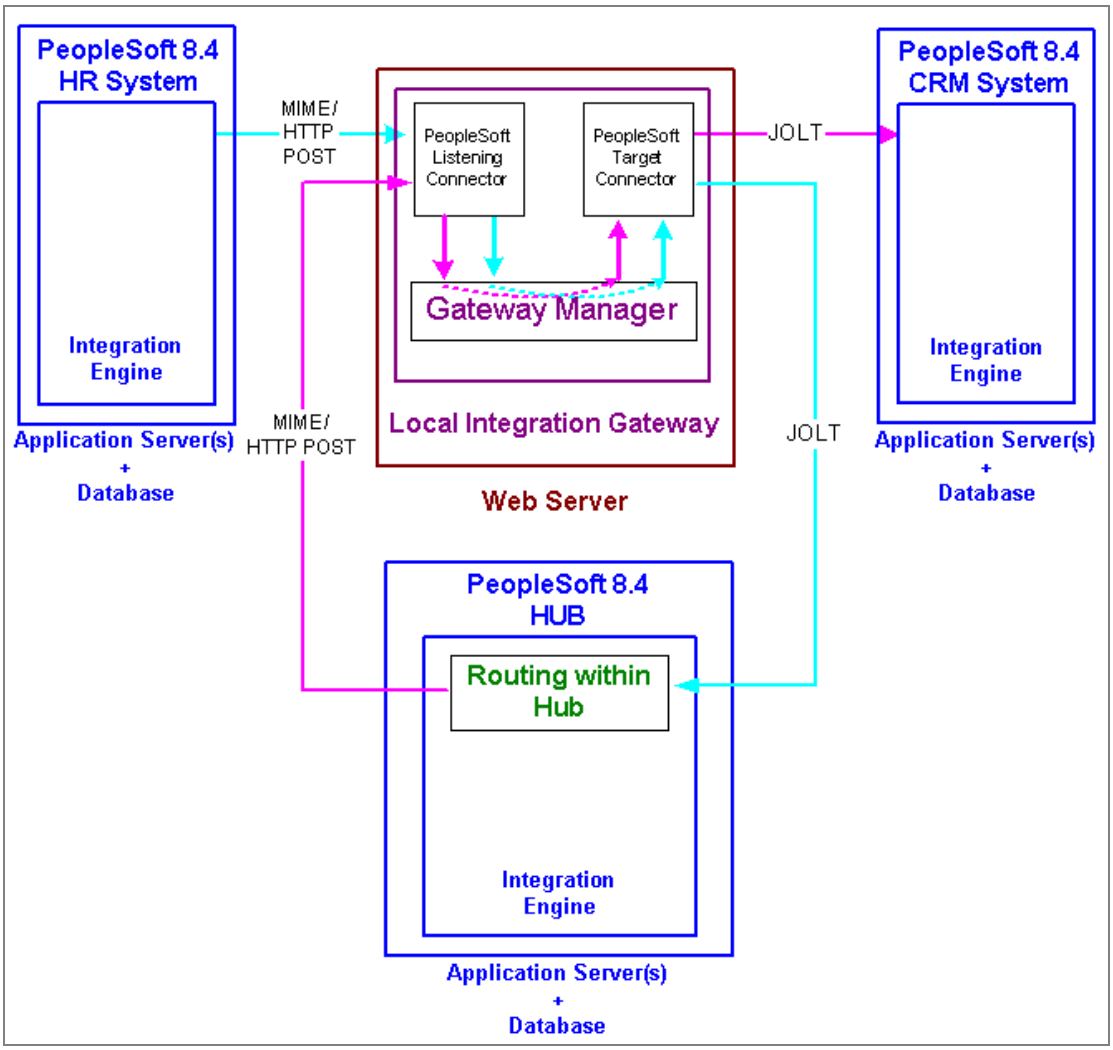
High-Level Configuration Steps

“Administering Basic Integrations” Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

“Administering Basic Integrations” Configuring Transactions

Integrations with PeopleSoft 8.4 Systems Using Hubs



Scenario: Integrations with PeopleSoft 8.4 systems using hubs.

A PeopleSoft Integration Broker Hub configuration consists of an Integration Engine that houses routing rules and transformations. All transactions are routed through the Hub, which allows you to centralize routing rules and offload the transformation process.

The diagram shows a Hub configuration scenario that involves a PeopleSoft HR system and a PeopleSoft CRM system. In this scenario, all of the routing rules (transactions) and transformations (relationships) are located on the Hub,

To implement integrations between these two systems without a Hub, you would have to set up a complete set of inverse routing rules and transformations on each node.

There are two Hub routing types, generic routing and sender-specific routing, described in the next section.

Regardless of which Hub routing you choose, you must configure each PeopleSoft 8.4 system, the Integration Gateway and the PeopleSoft 8.4 Hub. (Keep in mind that an Integration Broker Hub has only a stand-alone PeopleTools database installed, which includes the Integration Engine.)

Hub Routing Types

There are two Hub routing types, generic routing and sender-specific routing. The configuration steps for a Hub vary, depending on which routing type you choose.

With generic routing all transactions from the participating systems are sent directly to a Hub for routing and transformation.

With sender-specific routing a destination (target) node is passed as a parameter to a Publish or SyncRequest method, such as PublishXMLDoc or SyncRequestXMLDoc, to explicitly route the outbound transaction(s) to a node. Using sender-specified routing requires that you define the explicit destination nodes on the sending (source) system. However, you can configure your system so that PeopleSoft Integration Broker passes these outbound to the Hub for possible rerouting and transformation.

You must use sender-specific routing when you are using PublishXMLDoc to publish an XML object.

Generic-Routing Hub Configuration

Using the components shown in the diagram as an example, this section provides an overview of the configuration tasks to set up a generic-routing Hub.

As you review the configuration tasks, keep in mind that in the scenario shown, the PeopleSoft HR system, the PeopleSoft CRM system and Hub must all point to the same Integration Gateway and use the same gateway URL.

PeopleSoft HR System Configuration Tasks

On the PeopleSoft HR system:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.
- **Set up a local Node**. The local node you set up represents the HR system. Use the Node Definitions component to perform this task.
- **Set up a remote Node**. The remote node you set up represents the Hub system. Use the Node Definitions component to perform this task.
- **Set up outbound transactions**. Set up outbound transactions on the Hub (remote) node. Use the Transactions tab in the Node Definitions component to perform this task.

PeopleSoft CRM System Configuration Tasks

On the PeopleSoft CRM system:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.
- **Set up a local Node**. The local node you set up represents the PeopleSoft CRM system. Use the Node Definition component to perform this task.
- **Set up a remote Node**. The remote node you set up represents the Hub system. Use the Node Definition component to perform this task.
- **Set up inbound transactions**. Set up inbound transactions from the Hub node. Use the Transactions tab in the Node Definitions component to perform this task.

PeopleSoft Hub Configuration Tasks

On the PeopleSoft Hub:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.
- **Set up a local Node**. The local node you set up represents the Hub system. Use the Node Definition component to perform this task.
- **Set up remote Nodes**. Set up two remote nodes—one that represents the PeopleSoft HR system and another that represents the PeopleSoft CRM system.
- **Set up inbound transactions**. Create inbound transactions from the PeopleSoft HR system. Use the Transactions tab in the Node Definitions component to perform this task.
- **Set up outbound transactions**. Create outbound transactions for the PeopleSoft CRM system. Use the Transactions tab in the Node Definitions component to perform this task.
- **Set up relationships**. Set up relationships to link the inbound transactions from the PeopleSoft HR system to the outbound transactions for the PeopleSoft CRM system. Use the Relationships component to perform this task.

Integration Gateway Configuration Tasks

You must set Integration Gateway properties for the local gateway. The only required property you must set is the Jolt connect string(s) that enable communication with the Integration Engines on the PeopleSoft HR, PeopleSoft CRM and PeopleSoft Hub systems. Use the IntegrationGateway.properties file to set this property.

Generic-Routing Hub Configuration Summary

For all messages going through the Hub, you must set up transactions and relationships on the Hub. Using the systems in the diagram as an example, the following table shows the required node, transaction and relationship configurations required for generic routing through a Hub.

Item to Configure	*PeopleSoft 8.4 HR System*	*PeopleSoft 8.4 Hub*	*PeopleSoft 8.4 CRM System*
Local Nodes	Create local Node to represent HR system.	Create local Node to represent the Hub.	Create local Node to represent the CRM system
Remote Nodes	Create a remote Node to represent the Hub system.	Create remote Nodes to represent the HR and CRM systems.	Create a remote Node to represent the Hub
Transactions	Create outbound transactions to the Hub	Create inbound transactions from the HR system. Create outbound transactions for the CRM system.	Create inbound transaction from the Hub.
Relationships	N/A	Inbound PeopleSoft HR to outbound for CRM	N/A

See Also

High-Level Configuration Steps

"Administering Basic Integrations" Configuring a Gateway

"Administering Basic Integrations" Configuring a Node

"Administering Basic Integrations" Configuring Transactions

"Administering Relationships"

Sender-Specified Routing Hub Configuration

This section provides an overview of how to configure a Hub for sender-specific routing. Using the systems shown in the diagram as an example, the PeopleSoft 8.4 HR system is the sending system and the PeopleSoft 8.4 CRM system is the receiving system.

As you review the configuration tasks, keep in mind that for the scenario shown the sending system, the receiving system and the Hub must all point to the same gateway and use the same gateway URL.

PeopleSoft HR (Sending System) Configuration Tasks

In the previous diagram, the PeopleSoft 8.4 HR System is the sending system. On the PeopleSoft HR system:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.
- **Set up a local Node**. The local node you set up represents the HR system. Use the Node Definitions component to perform this task.

- **Set up remote Nodes**. Set up two remote nodes—one for the receiving system (PeopleSoft CRM in the example) and one for the Hub. When setting up the PeopleSoft CRM remote node, you must set the Routing Type to Explicit and in the Hub node field you must enter the node name of the Hub.

- **Set up outbound transactions**. Set up outbound transactions on the PeopleSoft CRM system. Use the Transactions tab in the Node Definitions component to perform this task.

PeopleSoft CRM (Receiving System) Configuration Tasks

In the previous diagram, the PeopleSoft 8.4 CRM system is the receiving system. On the PeopleSoft CRM system:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.

- **Set up a local Node**. The local node you set up represents the CRM system. Use the Node Definitions component to perform this task.

- **Set up a remote Node**. Set up a node that represents the Hub. Use the Node Definitions component to perform this task.

- **Set up inbound transactions**. Create inbound transactions from the Hub. Use the Transactions tab in the Node Definitions component to perform this task.

PeopleSoft Hub Configuration Tasks

On the PeopleSoft Hub:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.

- **Set up a local Node**. The local node you set up represents the Hub system. Use the Node Definitions component to perform this task.

- **Set up remote Nodes**. Set up two remote nodes—one for the PeopleSoft HR system and one for the PeopleSoft CRM system. Use the Node Definitions component to perform this task.

- **Set up inbound transactions**. Create inbound transactions from the PeopleSoft HR system. Use the Transactions tab in the Node Definitions component to perform this task.

- **Set up outbound transactions**. Create outbound transactions for the PeopleSoft CRM system. Use the Transactions tab in the Node Definitions component to perform this task.

- **Set up relationships**. Set up relationships for the PeopleSoft CRM system. These relationships will link inbound transactions from the PeopleSoft HR system to the PeopleSoft CRM system. Use the Relationships component to perform this task.

Integration Gateway Configuration Tasks

The only required Integration Gateway property you must set for the local Integration Gateway is the Jolt connect string(s) that enable communication to Integration Engines on destination PeopleSoft 8.4 systems. Use the IntegrationGateway.properties file to set this property.

Sender-Specified Routing Hub Configuration Summary

For all messages going through the Hub, you must set up transactions and relationships on the Hub. Using the systems in previous diagram as example, the following table shows the required Node, transaction and relationship configurations required for sender-specified routing through a Hub from the PeopleSoft HR system.

Item to Configure	*PeopleSoft 8.4 HR System*	*PeopleSoft 8.4 Hub*	*PeopleSoft 8.4 CRM System*
Local Nodes	Create local Node to represent HR system.	Create local Node to represent the Hub.	Create local Node to represent the CRM system.
Remote Nodes	Create remote Nodes to represent the CRM and HUB systems.	Create remote Nodes to represent the HR and CRM systems	Create a remote Node to represent the Hub.
Transactions	Create outbound transactions to CRM	Create inbound transactions from the HR system, and create outbound transactions from the CRM system.	Create inbound transaction from the Hub.
Relationships	N/A	Relationships will link inbound transactions from HR to outbound to CRM.	N/A

Additional Information

All messages to the CRM node will be the result of publish statements which include the target node parameters;

- msg.Publish(Node.CRM).
- SyncRequest(Node.CRM).
- PublishXMLDoc(&MyDoc, Message.MyMessage, Node.CRM) .
- SyncRequestXMLDoc(&MyDoc, Message.MyMessage, Node.CRM).

See Also

High-Level Configuration Steps

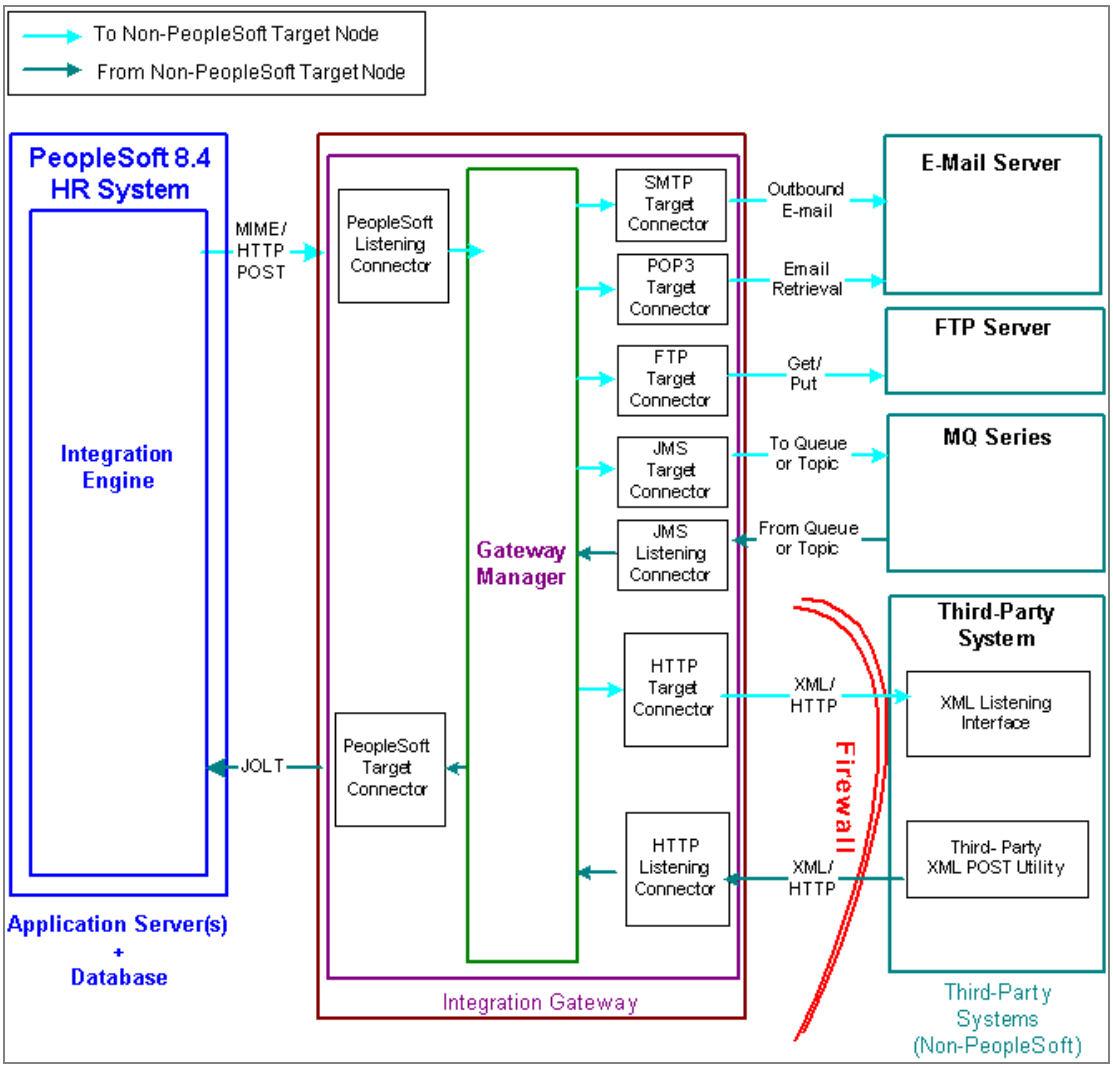
“Administering Basic Integrations” Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

"Administering Basic Integrations" Configuring Transactions

"Administering Relationships"

Integrations with Third-Party Systems



Scenario: Integrations with third-party systems using remote gateways.

For communications with third-party systems, messages can go through local or remote gateways.

Sending a message to a third-party system is the same as sending a message to a PeopleSoft 8.4 node, except that the target connector you select depends on the third-party system with which you are communicating.

Messages from third-party systems can enter the gateway via any of the listening connectors that PeopleSoft delivers with Integration Broker, or via a custom-built listening connector.

Note that you cannot use the PeopleSoft Listening Connector for integrations with third-party systems since it can only accept messages in the PeopleSoft internal format only.

The diagram shows the many connectors a PeopleSoft system can use to communicate with a third-party system. It also shows how the PeopleSoft system can communicate with third-party systems over a firewall.

Using the system components shown in the diagram as examples, the following sections highlight the configuration tasks for setting up communications between a PeopleSoft system and a third-party system.

PeopleSoft HR System Configuration Tasks

On the PeopleSoft HR system:

- **Define a local Integration Gateway**. Set up a local Integration Gateway for sending messages. Use the Gateways component to perform this task.
- **Set up a local node**. The local node you set up represents the HR system. Use the Node Definitions component to perform this task.
- **Set up a remote node**. Set up a remote node that represents the third-party system. When you define this node, you select the appropriate connector (for example, JMS Target Connector, POP3 Target Connector, SMTP Target Connector, and so forth) for communicating with the third-party system.
- **Set up inbound transactions**. Set up inbound transactions on the third-party (remote) node.
- **Set up outbound transactions**. Set up outbound transactions on the third-party (remote) node.

PeopleSoft HR Integration Gateway Configuration Tasks

The only required Integration Gateway property you must set for the local Integration Gateway is the default Jolt connect string(s) that enable communication to Integration Engines on the PeopleSoft 8.4 HR system. Use the IntegrationGateway.properties file to set this property.

See Also

High-Level Configuration Steps

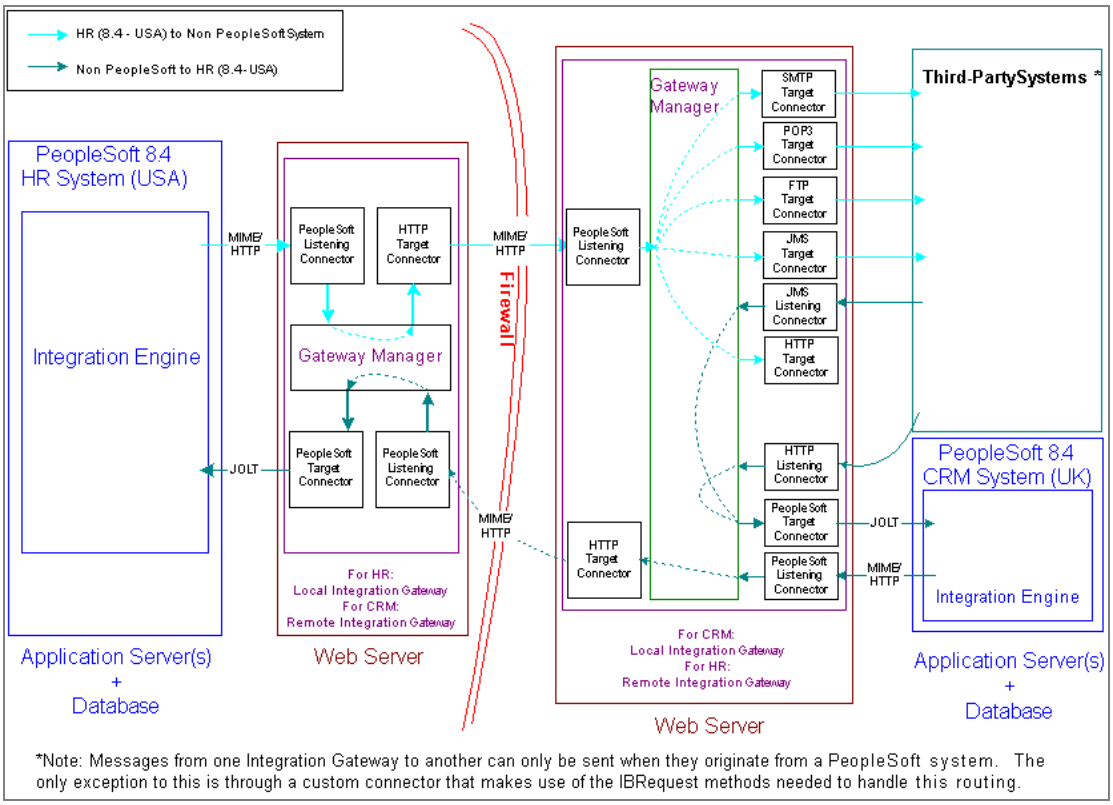
“Administering Basic Integrations” Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

“Administering Basic Integrations” Configuring Transactions

“Administering Relationships”

Integrations with Third-Party Systems Using Remote Gateways



Scenario: Integrations with third-party systems using remote gateways.

This section discusses:

- Sending messages to third-party systems using remote gateways.
- Sending messages from third-party systems to PeopleSoft systems using remote gateways.

Sending Messages to Third-Party Systems Using Remote Gateways

The set up for this scenario is similar to the configuration described in the scenario Understanding Integrations with PeopleSoft 8.4 Systems. The only difference is the target connector you use.

Instead of using the PeopleSoft Target Connector for the remote node you must select the target connector based on the third-party system with which the PeopleSoft system is communicating. The target connector you select must reside on the remote gateway.

Using the systems shown in the previous diagram as an example, the PeopleSoft 8.4 HR system is the source system and the selected target connector is shown on the other side of the firewall, on the remote Integration Gateway.

As you review the configuration tasks for this scenario, keep in mind the following points:

- PeopleSoft recommends using a single gateway for all applications that reside on one side of a firewall.
- The local gateway for one PeopleSoft application can be the remote gateway for the another PeopleSoft application.

PeopleSoft HR System Configuration Tasks

On the PeopleSoft HR system:

- **Define a local Integration Gateway**. Use the Gateways component to define the local HR (USA) gateway.
- **Define a remote Integration Gateway**. The remote gateway for the PeopleSoft HR (USA) system is the CRM (UK) gateway. Use the Gateways component to define the gateway, and specify the URL of the CRM (UK) gateway.
- **Set up a local node**. The local node you set up represents the HR (USA) system. Use the Node Definitions component to perform this task.
- **Set up a remote node**. The remote node you set up represents the third-party system. Use the Node Definitions component to perform this task. When you define the remote node, use the Connector tab to specify the gateway ID on the remote Integration Gateway. In addition, select the appropriate target connector, for example JMS Target Connector, SMTP Target Connector, POP 3 Target Connector, and so forth.
- **Set up inbound transactions**. Set up inbound transactions on the third-party (remote) node. This is only required if you will be receiving messages from this third-party system.
- **Set up outbound transactions**. Set up outbound transactions on the third-party (remote) node.

PeopleSoft HR Integration Gateway Configuration Tasks

The only required Integration Gateway property you must set is the Jolt connect string(s) that enable communication to Integration Engines on PeopleSoft 8.4 systems. Use the IntegrationGateway.properties file to set this property.

Third-Party Integration Gateway Configuration Tasks

Some connector-specific properties may also be required, please refer to the Gateway Configuration for more details. Use the IntegrationGateway.properties file to set this property.

See Also

High-Level Configuration Steps

"Administering Basic Integrations" Configuring a Gateway

"Administering Basic Integrations" Configuring a Node

"Administering Basic Integrations" Configuring Transactions

"Administering Relationships"

Sending Messages from Third-Party Systems to PeopleSoft 8.4 Systems

The diagram shows a second configuration scenario where a third-party system is performing an inbound HTTP Post to a PeopleSoft HR/USA system via a UK gateway. In this scenario, the message goes through the PeopleSoft CRM/UK system only to get routing information, before it is sent to the remote Integration Gateway (the gateway on the USA side of the firewall. Therefore, in this scenario, the PeopleSoft CRM/UK system serves as a hub.

Message Flow

In this scenario, a message originating from a third-party system is posted to the HTTP Listening Connector, JMS Listening Connector or a custom-built listening connector on the PeopleSoft CRM/UK Integration Gateway. Since the message does not contain the required routing information for the remote gateway, the listening connector hands it off to the PeopleSoft Target Connector. The PeopleSoft Target Connector sends the message to the default PeopleSoft node (the PeopleSoft CRM/UK) as determined by the default Jolt settings in the Integration Gateway properties file.

When the message reaches the Integration Broker on the PeopleSoft CRM/UK system, the system applies transaction information to reroute the message to the remote gateway (the gateway on the USA side of the firewall), and thereby serves as a hub.

Message Processing on the Remote Gateway

Whenever you publish a message bound for a remote gateway, the Integration Broker reads it, determines that the target connector is not on its local gateway, places the remote gateway URL inside the IBInfo message wrapper and posts it to the PeopleSoft Listening Connector on the local gateway. The local Gateway Manager finds a remote gateway URL in the message wrapper and routes it to the remote gateway default connector, the HTTP Target Connector. The HTTP Target Connector on the local gateway then posts it to the remote gateway URL (the PeopleSoft Listening Connector on the remote gateway) in MIME format, at the same time removing the URL from the IBInfo message wrapper. On arrival at the remote gateway, the message gets processed like any other incoming PeopleSoft message.

An exception to this message flow is if on the UK Integration Gateway side you created and loaded a custom listening connector that allows for the required routing information to be populated in the IBInfo message wrapper. The message would no longer need to be sent via the hub.

As you review the following configuration tasks for this scenario, keep in mind the following points:

- PeopleSoft recommends using a single gateway for all applications that reside on one side of a firewall.
- The local gateway for one PeopleSoft application can be a remote gateway for another PeopleSoft application.
- A message coming from a third-party system (local gateway or remote gateway) system can enter the Integration Gateway from any of the delivered listening connectors or from a custom-built listening connector. It cannot, however, use the PeopleSoft Listening Connector. PeopleSoft has designed the PeopleSoft Listening Connector to accept messages in the PeopleSoft internal message format only. Note that the diagram shows the message entering the Integration Gateway via the HTTP Listening Connector.

PeopleSoft HR (USA) System Configuration Tasks

On the PeopleSoft HR system:

- **Set up a local node**. The local node you set up represents the HR system. Use the Node Definition component to perform this task.
- **Set up a remote node**. The remote node you set up represents the CRM system. When you set up the remote node, specify the PeopleSoft Target Connector.
- **Set up inbound transactions**. Set up inbound transactions on the CRM (remote) node.
- **Set up outbound transactions**. Set up outbound transactions on the CRM (remote) node.

PeopleSoft HR Integration Gateway Configuration Tasks

The only required Integration Gateway property you must set is the Jolt connect string(s) that enable communication to PeopleSoft 8.4 HR systems. Use the IntegrationGateway.properties file to set this property.

PeopleSoft CRM (UK) System/Hub Configuration Tasks

In this scenario, the PeopleSoft CRM (UK) system serves as a hub. On the PeopleSoft CRM (UK) system:

- **Define a local Integration Gateway**. Using the Gateways Component set up the local Gateway for the sending system.
- **Define a remote Integration Gateway**. The remote gateway for the PeopleSoft CRM (UK) system is the PeopleSoft HR (USA) Gateway.
- **Set up a local node**. The local node you set up represents the CRM system. Use the Node Definition component to perform this task.

- **Set up remote nodes**. Define two remote nodes—one remote node that represents the third-party system and another to represent the PeopleSoft HR (USA system). Use the Node Definition component to perform this task.

When you define the remote node that represents the third-party system, you specify the HTTP Target Connector, HTTPTARGET.

When you define the remote node that represents the HR/USA system, set it to use the PeopleSoft Target Connector on the remote gateway (on the USA gateway).

- **Set up inbound transactions**. Set up inbound transaction on the third-party (remote) node. See the table at the end of this section for information about setting up transactions for this scenario.

- **Set up outbound transactions**. Set up outbound transactions on the HR (remote) node. See the table at the end of this section for information about setting up transactions for this scenario.

- **Set up outbound relationships**. These relationships link inbound transactions from the third-party system to outbound transactions to the HR system. Use the Relationships component to perform this task.

PeopleSoft 8.4 CRM (UK) Integration Gateway Configuration Tasks

The only required property you must set is the Jolt connect string(s) that enable communication to Integration Engines on PeopleSoft 8.4 CRM systems. Use the IntegrationGateway.properties file to set this property.

Third-Party System to PeopleSoft System Configuration Summary

Since the PeopleSoft CRM (UK) system serves as a hub in this scenario, you must set up transactions and relationships for all messages from the third-party system that get routed through it. Using the systems in the diagram as an example, the following table shows the required node, transaction and relationship configurations.

Item to Configure	*PeopleSoft 8.4 HR System*	*PeopleSoft 8.4 CRM System (Hub)*
Local Nodes	Create local node to represent HR system.	Create local node to represent the CRM system.
Remote Nodes	Create remote nodes to represent the CRM system.	Create remote nodes to represent the third-party system and the HR system.
Transactions	Create inbound transactions from the CRM system.	Create inbound transactions from the third-party system, and create outbound transactions to the HR system.
Relationships	N/A	Relationships will link inbound transactions from the third-party system to outbound to the HR system.

See Also

High-Level Configuration Steps

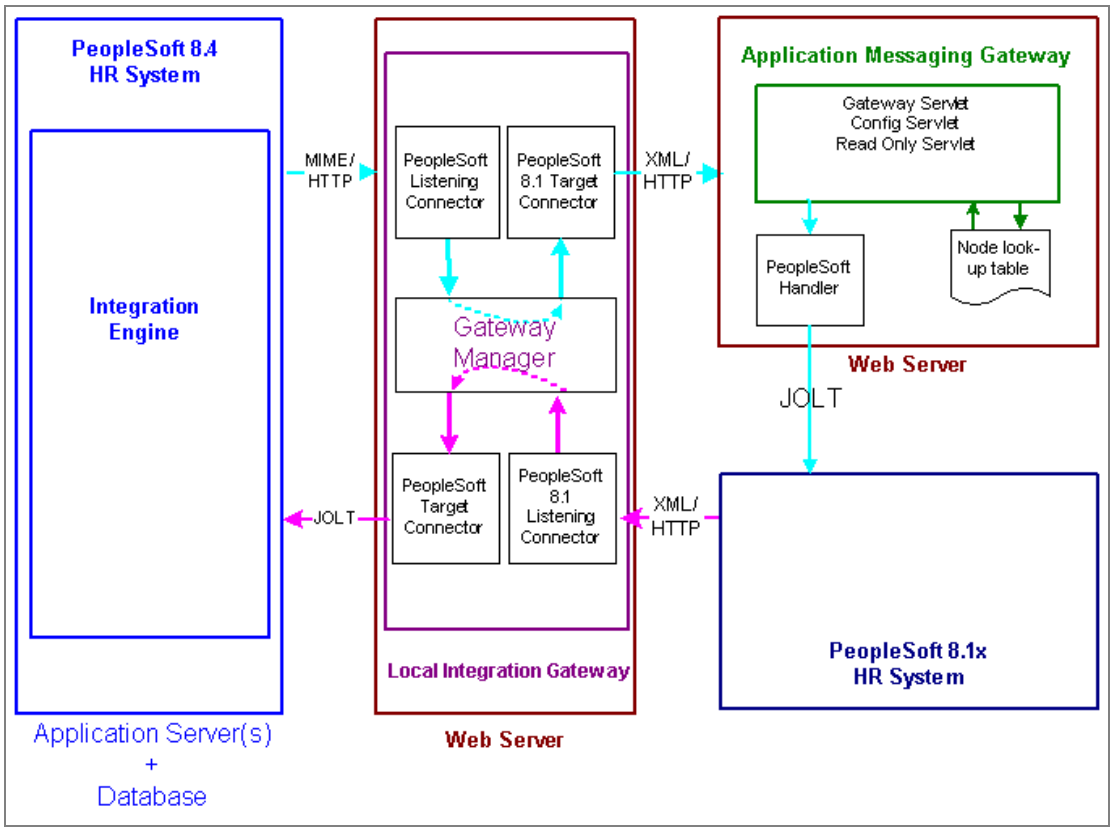
"Administering Basic Integrations" Configuring a Gateway

"Administering Basic Integrations" Configuring a Node

"Administering Basic Integrations" Configuring Transactions

"Administering Relationships"

Integrations with PeopleSoft 8.1 Systems



Scenario: Integrations with PeopleSoft 8.1 Systems

The previous diagram shows the Integration Broker components and configuration for communications between PeopleSoft 8.4 and PeopleSoft 8.1 systems.

In this scenario you must configure the PeopleSoft 8.4 system, the Integration Gateway and the PeopleSoft 8.1 system. The remainder of this section highlights these tasks, using the systems and components shown in the diagram as examples.

PeopleSoft 8.4 HR System Configuration Tasks

On the PeopleSoft 8.4 HR system:

- **Define a local Integration Gateway**. Using the Gateways Component set up a local gateway for the HR system.
- **Set up a local node**. The local node you set up represents the PeopleSoft 8.4 HR system. Use the Node Definition component to perform this task.
- **Set up a remote node**. The remote node you set up represents the PeopleSoft 8.1 HR system. When you set up the remote node specify the PeopleSoft 8.1 Target Connector (PSFT81TARGET) on the Connectors tab.

Note. If you have upgraded from a PeopleSoft 8.1 system, all nodes that existed for the system have been preserved as remote nodes in the PeopleSoft 8.4 system. However, you must then associate each of these nodes to the PeopleSoft 8.1 Target Connector .

- **Set up inbound transactions**. Set up inbound transactions on the PeopleSoft 8.1 HR (remote) node.
- **Set up outbound transactions**. Set up outbound transactions on the 8.1 HR (remote) node.

PeopleSoft 8.4 HR Integration Gateway Configuration Tasks

You must set Integration Gateway properties for the local gateway. The only required property you must set is the Jolt connect string(s) that enable communication to PeopleSoft 8.4 HR systems. Use the IntegrationGateway.properties file to set this property.

PeopleSoft 8.1 HR System Configuration Tasks

On the PeopleSoft 8.1 HR system, locate the 8.1 HR Message Node and change the URL (Location) to the PeopleSoft Listening Connector. The format is:
http://machinename/PSIGW/PS81ListeningConnector

See Also

High-Level Configuration Steps

“Administering Basic Integrations” Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

“Administering Basic Integrations” Configuring Transactions

“Administering Relationships”

High-Level Configuration Steps

This section provides more detailed overview information about the configuration tasks that apply to most, if not all, of all the configuration scenarios presented in this chapter.

About Defining Local and Remote Integration Gateways

On each PeopleSoft 8.4 systems in your configuration, you need to specify the default local gateway. The local Integration Gateway is the application's first point of contact with other PeopleSoft applications, third-party systems, Integration Broker Hubs, and remote gateways. Integration Broker requires that you define exactly one local gateway on each PeopleSoft 8.4 system.

To add the default local gateway, you use the Gateways component. When you add the default local gateway, you perform the following actions:

- Add the gateway that your application will use to communicate with other systems.
- Set the URL to the PeopleSoft Listening Connector. This is the connector that receives incoming messages from an Integration Engine or remote gateway.
- Load target connectors bundled with Integration Broker.

The Web server on which the Integration Gateway resides can be on either of the physical machines on which you have installed the PeopleSoft 8.4 applications, or it can reside on its own machine.

Regardless of where the Integration Gateway resides, keep in mind that the Integration Gateway can serve multiple application servers. When you define the local Integration Gateway for each application server, you specify the same URL for each application server.

As mentioned previously, when you define the local Integration Gateway, you also load the target connectors that PeopleSoft bundles with the Integration Broker. These connectors are automatically installed during the PeopleSoft Internet Architecture (PIA) install process. Later, when you define local and remote nodes, you will specify which target connector to use for the configuration.

Note. The remote gateway default connector setting in the IntegrationGateway.properties file, ig.connector.defaultremoteconnector, determines to which connector a local gateway routes messages that are bound for remote gateways. By default this property is set to the HTTP Target Connector, HTTPTARGET. Never change this setting unless you develop a custom connector to handle this routing.

PeopleSoft also bundles listening connectors with Integration Broker. They are installed and loaded during the PIA installation process. You do not to need to define or specify these connectors.

About Setting Integration Gateway Properties

You set Integration Gateway properties using the IntegrationGateway.properties file. This file is located on the Web server where the Integration Gateway resides.

The only required property you must set is the Jolt connect string(s) that enable communication to other PeopleSoft Integration Engines. You must define Jolt strings for all PeopleSoft nodes with which you wish to integrate.

The following example shows the properties you must specify.

```
ig.isc.$NODENAME.serverURL=//<machine name>:<jolt port>
ig.isc.$NODENAME.userid=<database user id>
ig.isc.$NODENAME.password=<database password>
ig.isc.$NODENAME.toolsRel=<peopletools release version>
```

About Setting Up Local and Remote Nodes

You use the Node Definition component to set up local and remote nodes.

Note. If you have upgraded from a PeopleSoft 8.1 system, all nodes that existed for that system have been preserved as remote nodes. You must assign each of them an appropriate target connector.

When you set up local and remote nodes, use the Connectors tab in the Node Definition component to specify a target connector for each node. Use the information in the following table as a guide for choosing the appropriate information for the configuration scenarios described in this chapter.

Scenario	*System*	*Node*	*Connector*
PeopleSoft 8.4 to PeopleSoft 8.4	PeopleSoft 8.4	Local	N/A.*
		Remote	PSFTTARGET (PeopleSoft Target Connector)
PeopleSoft 8.4 to PeopleSoft 8.4 Using a Remote Gateway	PeopleSoft 8.4	Local	N/A
		Remote	PSFTTARGET (PeopleSoft Target Connector)
PeopleSoft 8.4 to PeopleSoft 8.4 Using a Hub	PeopleSoft 8.4	Local	N/A.*
		Remote	PSFTTARGET (PeopleSoft Target Connector)
	Hub	Local	N/A.*
		Remote	PSFTTARGET (PeopleSoft Target Connector)
PeopleSoft 8.4 to Third-Party	PeopleSoft 8.4 System	Local	N/A.*

Scenario	*System*	*Node*	*Connector*
		Remote	Third-Party Connector as appropriate: • JMSTARGET (JMS Target Connector) • POP3TARGET (POP3 Target Connector) • SMTPTARGET (SMTP Target Connector) • And so forth.
	Third-Party System	N/A	N/A
PeopleSoft 8.4 to Third-party Using Remote Gateway	PeopleSoft 8.4 System (Each system)	Local	N/A *
		Remote	Third-Party Connector as appropriate: • JMSTARGET (JMS Target Connector) • POP3TARGET (POP3 Target Connector) • SMTPTARGET (SMTP Target Connector) • And so forth.
	Third-Party System	N/A	N/A
PeopleSoft 8.4 to PeopleSoft 8.1 or **PeopleSoft 8.1 to PeopleSoft 8.4**	PeopleSoft 8.4 System	Local	N/A
		Remote	PSFT81TARGET (PeopleSoft 8.1Target Connector)
	PeopleSoft 8.1 System	Local	N/A. Set up message nodes, message channels, messages, and so on.

* Defaults to PSFTTARGET (PeopleSoft Target Connector) but is not used.

See Also

"Using the Integration Gateway" Setting Integration Gateway Properties

"Administering Basic Integrations" Configuring a Gateway

“Administering Basic Integrations” Configuring a Node

“Administering Basic Integrations” Configuring Transactions

“Administering Relationships”

CHAPTER 3

Creating and Implementing Integrations

This chapter is your guide to the process of producing integrations with Integration Broker. It directs you to more specific information about Integration Broker's features, and explains how to:

- Establish the integration environment.
- Produce basic integrations.
- Produce advanced integrations.

Establishing the Messaging Environment

Before you begin creating an integration, you must set up the Integration Broker messaging environment. Make sure you've completed the following steps:

1. Determine the purpose and requirements of the integration you want to develop.
2. Determine which PeopleSoft applications and third party systems (nodes) are going to participate in your integration.
3. Determine the nature of each node's role in the integration, as a sender, receiver or hub (or all three).
4. Install the local and remote integration gateways necessary to support each node.
5. Configure the connectors associated with each gateway.

It's possible that one local gateway may be sufficient for all your integrations. See "Understanding Integrations" for information about Integration Broker's gateway and node configuration options. See "Using the Integration Gateway" for details about setting gateway properties and configuring connectors.

Producing Basic Integrations

This section helps you to:

- Understand the basic messaging elements
- Understand the messaging process flow

Understanding the Basic Messaging Elements

The process of implementing any messaging integration, simple or complex, always starts with the same initial steps. The simplest integration requires no additional steps, and may be all you need. It has two distinct phases — development and administration.

Development Elements

Develop the basic elements of an integration. Once developed, they can be reused in a variety of different integrations.

1. *Message channel definition:* A channel isolates different groups of messages from each other. Messages in the same channel are processed sequentially. Each participating node must contain a channel definition by the same name. See "Defining Message Channels."

2. *Message definition:* This is the core element, which holds the message data. Each participating node must contain a message definition by the same name, with the same structure. Use the default message version, populate it with database records, and assign it to the channel you defined. See "Defining Messages."

3. *Sending or receiving PeopleCode:* PeopleCode that sends a message builds and populates it from the message definition, and can be launched from several places in your application. PeopleCode that receives the message must be associated with a message *subscription* (asynchronous) or the message *OnRequest* event (synchronous). See "Sending and Receiving Messages."

Note. If this is a synchronous integration, you must provide a response message definition in addition to the request message definition.

Administrative Elements

Set up and administer the definitions that determine how the basic elements will be used in an integration. Each participating Integration Broker node must contain at least one of each of these definitions.

1. *Gateway definition:* An application's internal representation of an installed Integration Gateway. Your application requires at least the *local gateway*, through which it can send and receive messages. Multiple nodes can share the same local gateway, which may be the only gateway you'll need for all your integrations. See "Administering Basic Integrations," Configuring a Gateway.

2. *Node definition:* An application's internal representation of a system with which it exchanges messages. Because your application can send messages to itself, a *default local node* definition that represents your application is delivered as part of Integration Engine. See "Administering Basic Integrations," Configuring a Node.

3. *Transaction definition:* A transaction assembles the other elements in combinations that form the basis of a specific integration, by specifying a request message definition and version, a transmission type (asynchronous or synchronous), and a direction (outbound from or inbound to the default local node). It also specifies the other node involved by

virtue of being part of that node's definition. See "Administering Basic Integrations," Configuring Transactions.

Understanding the Messaging Process Flow

Whether the Integration Engine is dealing with outbound or inbound messages, its primary agents for managing the messages are the *asynchronous request handler* and the *synchronous request handler*. These handlers work behind the scenes, examining the definitions you created in the development and administration phases to determine how each message should be treated.

Basic Outbound Messaging

1. Your application triggers the sending PeopleCode you've developed.
2. Your PeopleCode program populates and sends the message using a synchronous or asynchronous method or function.
3. The method used triggers either the asynchronous request handler or the synchronous request handler in your application's Integration Engine.
4. The handler searches the outbound transaction definitions associated with that message definition to determine the valid target nodes for the message.

   The asynchronous handler examines only asynchronous transactions and the synchronous handler examines only synchronous transactions. If no node definition that contains an outbound transaction specifying that message is found, the sending method returns an error.

   Note. Any outbound transaction found, the message to which it applies, and the node for which it's defined must all have an active status for the handler to proceed.

5. For each outbound transaction found, the handler submits the message to the local gateway, along with transaction information about the target node, and the target connector that should be used to send the message.
6. The local gateway transmits the message to the specified target node through the specified target connector.
7. If this is a synchronous message, the handler waits for the target node to pass a response message back through the gateway, then returns it to the calling PeopleCode method or function.

Basic Inbound Messaging

1. Your application's local gateway receives a request message from a remote node or gateway, which specifies your application as its target node, and indicates whether the message is to be processed asynchronously or synchronously.

2. The local gateway submits the message to your application's Integration Engine, which searches the inbound transaction definitions to find one associated with the sending node that specifies the same message, version and transmission type.

3. If no matching transaction is found, Integration Engine returns an error message through the gateway to the sending node. If a transaction is found, Integration Engine invokes either the asynchronous request handler or the synchronous request handler as appropriate to handle the message.

Note. Any inbound transaction found, the message to which it applies, and the node for which it's defined must all have an active status for the handler to proceed.

---

4. The handler accesses the message definition that matches the inbound message name and passes the message to its associated receiving PeopleCode.

   - The Asynchronous Request Handler invokes the message definition's subscription PeopleCode.

   - The Synchronous Request Handler invokes the message definition's OnRequest PeopleCode.

5. If this is a synchronous transaction, the handler waits for the receiving PeopleCode to generate and return a response message, then passes it back to the sending node through the gateway.

Producing Advanced Integrations

You can apply a number of variations to a basic integration to accommodate a wide variety of specialized messaging needs. This section outlines what you can do to:

- Enhance basic development and administration.

- Apply transaction modifiers to your integration.

- Invoke transform programs that modify your message.

Enhancing Basic Development and Administration

Following are some of the enhancements you can apply to your messages in the context of the basic development process:

- *Define multiple message versions.* Versions are variations of a message structure that nonetheless exist for the same business purpose and are part of the same message definition. Different nodes may require different message versions. See "Defining Messages," Defining a Message Version.

- *Employ nonrepudiation.* This security feature ensures that each node participating in a transaction cannot deny having sent or received the message, because the other participating nodes possess proof of the transaction. Nonrepudiation requires a setting in the message definition and the node definition. See "Defining Messages," Configuring Message Properties.

- *Send non-rowset based XML messages.* This refers to messages not defined using database records, but still compliant with the World Wide Web Consortium's (W3C) XML Document Object Model (XML DOM). You use the PeopleCode XmlDoc class to generate, populate and establish the structure of the message. See "Sending and Receiving Messages," XML Document Object Model.

- *Send SOAP compliant XML messages.* This is a specific variation of non-rowset based XML, compliant with the W3C's specification for the Simple Object Access Protocol (SOAP) for distributed synchronous transactions. You use the PeopleCode SoapDoc class to generate, populate and establish the structure of the message. See "Sending and Receiving Messages," Simple Object Access Protocol.

- *Send non-XML messages.* You can send virtually any kind of data in a message; however, you must enclose the data in an XML wrapper. See "Sending and Receiving Messages," Generating and Sending a Message.

- *Handle cookies in messages.* Integration Broker enables your application to extract cookies sent by a target node as part of a synchronous response message, and return them in a later message. You can handle cookies only in rowset-based messages. See "Sending and Receiving Messages," Handling Cookies.

- *Enable message authentication.* You can have Integration Broker authenticate messages using digital certificates or passwords. Authentication requires a setting in the node definition. Digital certificates must also be set up in the PeopleTools Security components. See "Administering Basic Integrations," Specifying General Node Information.

Applying Transaction Modifiers

Every integration requires at least one transaction defined at each participating Integration Broker node: One node uses a transaction to send a message, and one or more nodes use transactions to receive the message. If the the transactions at the sending and receiving nodes are defined with different parameters, they'll be incompatible, and the message won't be successfully transmitted.

You can apply a *transaction modifier* to the initial transaction, producing a resulting transaction compatible with the one used by the target node.

You define a transaction modifier with respect to a specified pair of nodes, which is called a *relationship.* All of the transaction modifiers that apply to a given node pair are part of the same relationship definition.

A transaction modifier can directly handle the following:

- *Change the transmission type:* The source node is sending a message asynchronously, but the target node expects a synchronous message. The transaction modifier forwards the asynchronous message synchronously, accepts the synchronous response, and returns it to the source node as another asynchronous request. See "Administering Relationships," Determining Relationship Parameters and "Administering Relationships," Managing Transaction Modifiers.

- *Route the message through a hub node:* If the receiving node is not the message's ultimate destination, it's by definition a *hub node*. You implement hub routing by applying a transaction modifier that changes an inbound transaction to an outbound transaction — the message that came in from the source node goes back out to the ultimate target node (or to another hub). See "Administering Relationships," Determining Relationship Parameters and "Administering Relationships," Managing Transaction Modifiers.

- *Retain the message at the hub node:* In addition to routing a message between two other nodes, the hub node may be an application that needs to consume the message as well. See "Administering Relationships," Retaining Messages at a Hub Node.

Invoking Transform Programs

A transaction modifier doesn't directly alter the content of a message, but you can invoke a *transform program* which does. A transform program is a type of Application Engine program that you develop and specify as part of a transaction modifier. Integration Broker supports the use of Extensible Stylesheet Language Transformation (XSLT) code and PeopleCode for developing transform programs.

Transform programs provide the following capabilities:

- *Transformation:* You apply a transformation to a message to make its structure comply with the target system's requirements. A transformation may be needed when one node is sending a request or response message with a data structure different from the structure required by the other node. See "Applying Transformation, Translation and Filtering," Applying Transformations.

- *Data translation:* You perform a data translation on a message so its content is represented according to the target system's conventions. It's appropriate when the sending and receiving systems use different field values, or different combinations of fields and their values, to represent the same information. Data translation relies on a repository of *codeset metadata* that you define. This means you can establish consistent rule-based translations, and reuse the same data without having to re-enter it. See "Applying Transformation, Translation and Filtering," Performing Data Translation.

- *Filtering:* You can programmatically determine whether to pass a message through to its target, by *filtering* it based on its content. You use PeopleCode to examine the message for criteria you define. If the message fails to meet your criteria, you strip out the message content and set a status flag informing Integration Broker of the failure. See "Applying Transformation, Translation and Filtering," Filtering Messages and Generating Errors.

Note. To develop a transform program, you must know in detail the initial structure and possibly the content of the message you're working with, as well as the structure (and content) of the result you want to achieve. See "Applying Transformation, Translation and Filtering," Understanding the PeopleSoft Base Message Format for details about PeopleSoft's standard message structure.

CHAPTER 4

Defining Message Channels

This chapter explains how to:

- Define a new message channel.
- Assign messages to the channel.
- Configure the message channel.
- Receive a subset of messages.

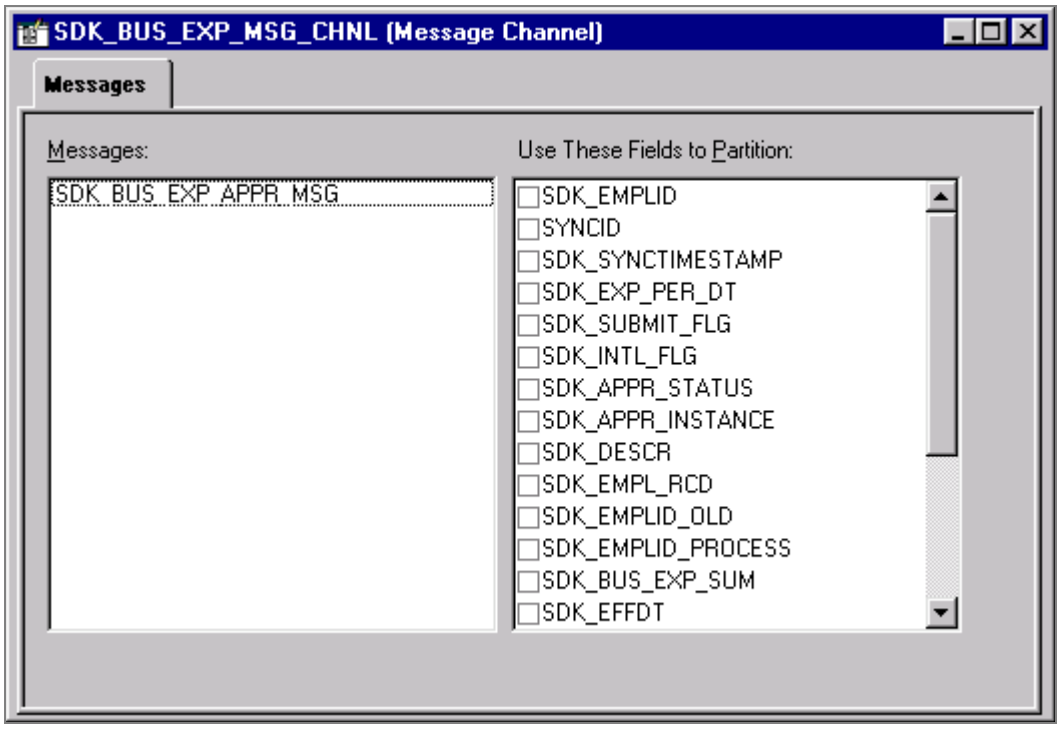
Understanding Message Channels

Channels are a logical grouping of messages. Each message must belong to one and only one message channel. Channels are used for the following:

- Establishing message processing sequence.
- Increasing throughput by using unordered processing.
- Maintaining messaging security.
- Maintaining messages with common configuration properties.
- Defining the processing attributes for timeout parameters, error thresholds, and delivery type.

Every message must be assigned to a channel, which you must define before you can define the message. You create channel definitions in Application Designer.

Defining a New Message Channel



Message channel definition

To define a new message channel:

1. In Application Designer, select File, New, Message Channel.

   A new channel definition appears. It consists of two main views—the messages list on the left, and the partitioning field list on the right.

   Note. In a new channel definition, both views are blank. You can name the channel and set its properties, but you must define messages in Application Designer before you can assign them to the channel or partition the channel. You can find more information about partitioning on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

   ---

2. Save the message channel.

   If this is a new channel, you must save it before you can add messages. You'll be prompted for an appropriate channel name.

Assigning Messages to the Channel

Before you can assign a message to your message channel, the message definition must already exist. You make the assignment in the message definition's properties.

When you create a message or open a previously created message, open the message properties dialog box for the message you want to add, select the Use tab, specify the message channel you want to assign it to in the Message Channel field, and save the message.

Opening a Message Definition

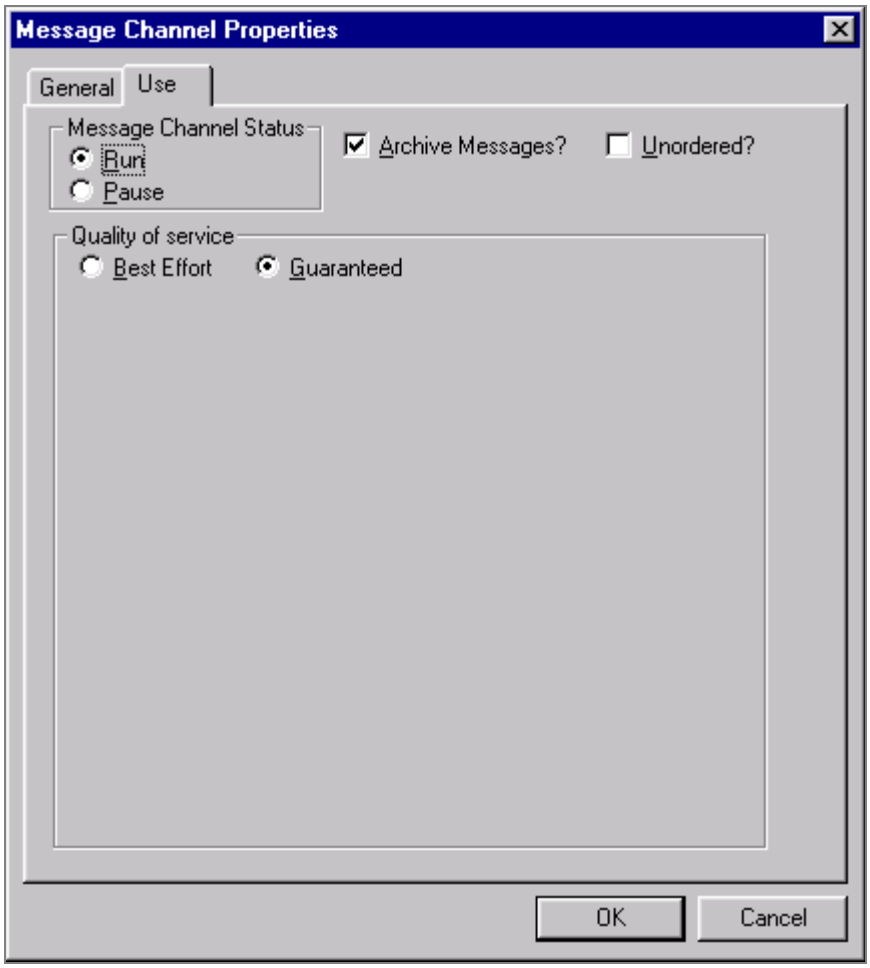
When you highlight the message channel definition, its assigned messages automatically display in the Messages list on the Messages tab. To open a message definition directly from this list, double-click the name of the desired message.

See Also

"Defining Messages"

Configuring the Message Channel

Access the message channel properties dialog box, and select the Use tab.



Message channel properties dialog box — Use tab

Message Channel Status

Select from the following values:

Run: Messages assigned to this channel are received and processed normally.

Pause: Messages are received but not processed until the status is reset to *Run.*

Note. Message channels may also be paused and restarted with Integration Broker Monitor. See "Using Integration Broker Monitor," Monitoring Channel Status.

Archive Messages?

Select this check box to have instances of messages assigned to this channel archived for safekeeping. If you don't select this check box, the messaging archive process purges the queue entries that have been processed.

This check box also controls whether the Archive or Delete action is available in the Message Details

component of the Integration Broker Monitor. See "Using Integration Broker Monitor," Monitoring Asynchronous Message Details.

Unordered?

By default, inbound messages assigned to this channel are processed one at a time, in the order they were sent. This means the sending node can engage in a series of transactions that modify a specific record, with assurance that they'll be applied by the receiving node in the correct order. This can be rather inefficient when the sequence doesn't matter.

Select this check box to force the channel to handle all of its messages in parallel (unordered). With this option, you can't assure any particular processing sequence. This disables the channel's partitioning fields.

Clear this check box if you want all of the channel's messages processed sequentially, or if you want to use the partitioning fields to specify which messages to process sequentially. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Quality of Service

The *Best Effort* option is not currently implemented. Quality of service is always *Guaranteed:* If Integration Broker fails to deliver a message, it retries until the time-out period expires. Then it marks the message as *Timeout* in Integration Broker Monitor. Once the target system is ready to receive the message, the system administrator can resubmit any message that timed out.

Deleting a Channel During Upgrade

To delete a channel definition in an application upgrade project, you must first make sure there are no live instances of messages assigned to that channel. Archive or delete any such messages in both the source and the target database. Otherwise, you'll see an error message during the copy process saying the message is in use. See "Using Integration Broker Monitor" for information about archiving and deleting messages.

Message Processing Order

When you clear the Unordered check box, Integration Broker guarantees that messages in each partition are delivered in the order sent and that they're single-threaded at the PeopleSoft receiving node. However, specific message order is not part of the channel definition, even when you apply partitioning. Therefore, it's your responsibility to send messages in the proper order.

For example, you can't specify in the channel definition that a message named PO_CREATE be processed before a message named PO_UPDATE. If you send PO_UPDATE before PO_CREATE, the recipient processes PO_UPDATE first.

Receiving a Subset of Messages

When a channel definition on the receiving node contains a subset of the messages assigned to the matching channel definition on the sending node, the receiving node receives some messages for which it has no definitions. How it handles this discrepancy depends on a setting in the configuration file for the current PeopleSoft application server domain, PSAPPSRV.CFG.

In the PSAPPSRV section of PSAPPSRV.CFG, the setting is expressed by:

```
Ignore Undefined Subscription Messages=
```

- If Ignore Undefined Subscription Messages=**1**, the receiving node will do just that—ignore any incoming message for which the receiving channel has no definition. This is the default setting.
- If Ignore Undefined Subscription Messages=**0**, the receiving node will reply to the sending node with an exception when it encounters an undefined message, which prevents any more messages from being sent until the exception is handled.

You can change this setting using the PeopleSoft Server Administration utility, PSADMIN, or by editing the configuration file or one of the templates on which it's based.

See Also

PeopleTools PeopleBook: Internet Architecture Administration, "Working with PSADMIN"

CHAPTER 5

Defining Messages

This chapter explains how to:

- Create message definitions.
- Define a message version.
- Use records in a message definition.
- Configure message properties.
- Access message PeopleCode.

Understanding Messages

The message definition is the heart of PeopleSoft Integration Broker. It serves as a template and reference point for the data your application sends or receives. You create message definitions in Application Designer.

- *Rowset based messages:* For hierarchical data based on PeopleSoft records, you create a message definition by assembling records, organizing them into a desired hierarchy, and selecting fields from those records to include in the message. The result is a rowset that you design, which isn't required to match any existing rowset structure in your application. Use the PeopleCode Rowset and Message classes to generate, send, receive and process these messages.
- *Non-rowset based messages:* You can also send and receive messages with virtually any other content and structure. You still create a message definition, but without inserting any records. The message definition is unstructured, and serves as a placeholder for the actual message. Use the PeopleCode XmlDoc class to generate, send, receive and process these messages. If you're handling SOAP-compliant data, you can also use the SoapDoc class to generate and process these messages.

For asynchronous integrations, define a single message. For synchronous integrations, define two messages — one for the request, and one for the response.

See Also

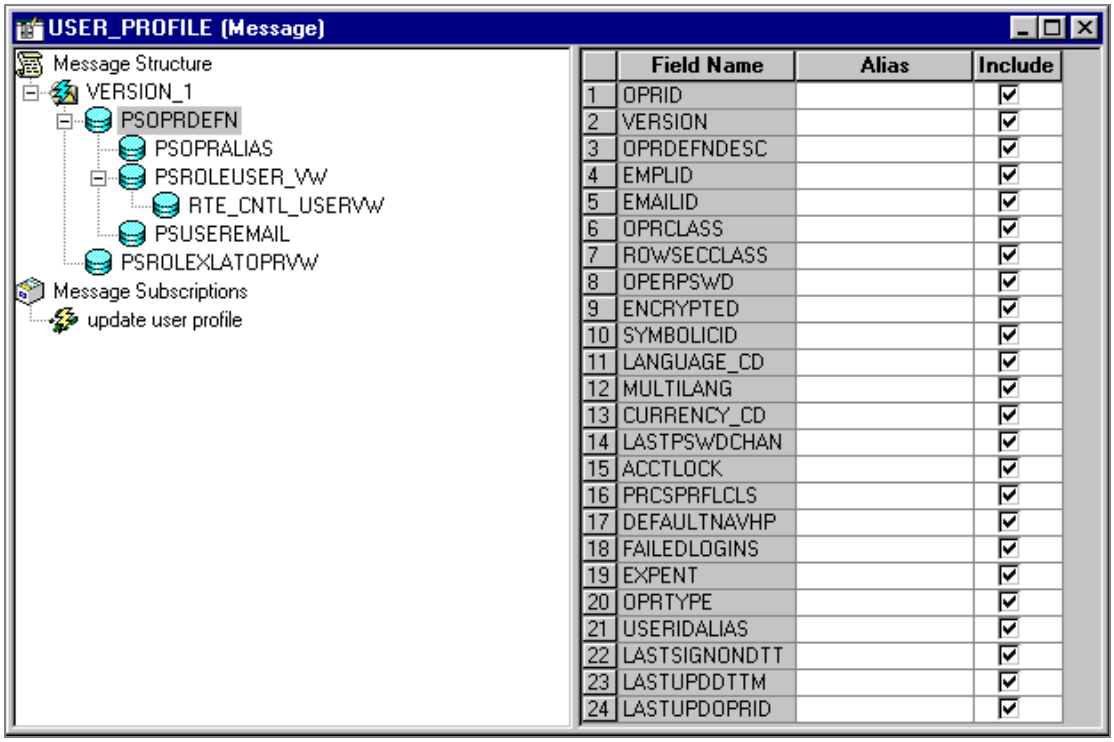
"Sending and Receiving Messages"

Creating Message Definitions

This section explains how you can:

- Define a new message.
- Understand the message definition.

Defining a New Message



	Field Name	Alias	Include
1	OPRID		☑
2	VERSION		☑
3	OPRDEFNDESC		☑
4	EMPLID		☑
5	EMAILID		☑
6	OPRCLASS		☑
7	ROWSECCLASS		☑
8	OPERPSWD		☑
9	ENCRYPTED		☑
10	SYMBOLICID		☑
11	LANGUAGE_CD		☑
12	MULTILANG		☑
13	CURRENCY_CD		☑
14	LASTPSWDCHAN		☑
15	ACCTLOCK		☑
16	PRCSPRFLCLS		☑
17	DEFAULTNAVHP		☑
18	FAILEDLOGINS		☑
19	EXPENT		☑
20	OPRTYPE		☑
21	USERIDALIAS		☑
22	LASTSIGNONDTT		☑
23	LASTUPDDTTM		☑
24	LASTUPDOPRID		☑

User Profile message definition

To define a new message:

1. In Application Designer, select File, New, Message.

   A new message definition appears. It consists of two main views—the *message structure* on the left, and the *field list* on the right.

   Note. Initially, the message structure just contains placeholder entries for the three primary elements of the message. See Understanding the Message Definition.

2. Assign the message to a message channel.

Open the message properties dialog box for your new message definition, select the Use tab, specify the message channel you want to assign it to in the Message Channel field, and click OK.

Note. You can't save your message definition until you assign it to an existing message channel. The message channel you use must already exist. See "Defining Message Channels" for more information.

3. Save the message definition.

   You'll be prompted for an appropriate message name.

Deleting a Message During Upgrade

To delete a message definition in an application upgrade project, you must first make sure there are no live instances of that message. Archive or delete any such messages in both the source and the target database. Otherwise, you'll see an error message during the copy process saying the object is in use. See "Using Integration Broker Monitor" for information about archiving and deleting messages.

Understanding the Message Definition

A message definition consists of the message structure view and the field list view.

Message Structure View

The message structure view shows several elements of the message definition:

- The *Message Structure* section displays different versions of the message, and the hierarchy of record definitions within each version. To use the message definition you must define at least one version; new message definitions supply an initial entry called VERSION_1 by default. If you plan to use the message definition to send or receive PeopleSoft rowset compatible data, you must populate the version with record definitions. See Defining a Message Version.
- The *Message Subscriptions* section contains the PeopleCode programs used to receive and process inbound asynchronous messages. See Creating a Message Subscription.

Note. To receive and process inbound synchronous messages, your PeopleCode must be associated with the message OnRequest PeopleCode event. See Accessing and Modifying OnRequest PeopleCode.

Field List View

The field list view displays in a grid the fields comprising the currently selected record definition in the message structure. If no record definition exists or none is selected, the grid is empty. The field list grid includes the following columns:

- The *Field Name* column lists all of the fields in the selected record definition.
- The *Alias* column enables you to specify for each field an XML field tag value which differs from the original field name. The Integration Broker sends the field with this name instead of the original field name on the sending system. If the sender doesn't specify an alias value, the field is sent with the original field name. The Alias field is useful for cross release and third party integration.
- The *Include* column enables you to specify which fields from the selected record will be sent or received. Select the check box to send the field or to expect it in an inbound message. Clear it to ignore the field.

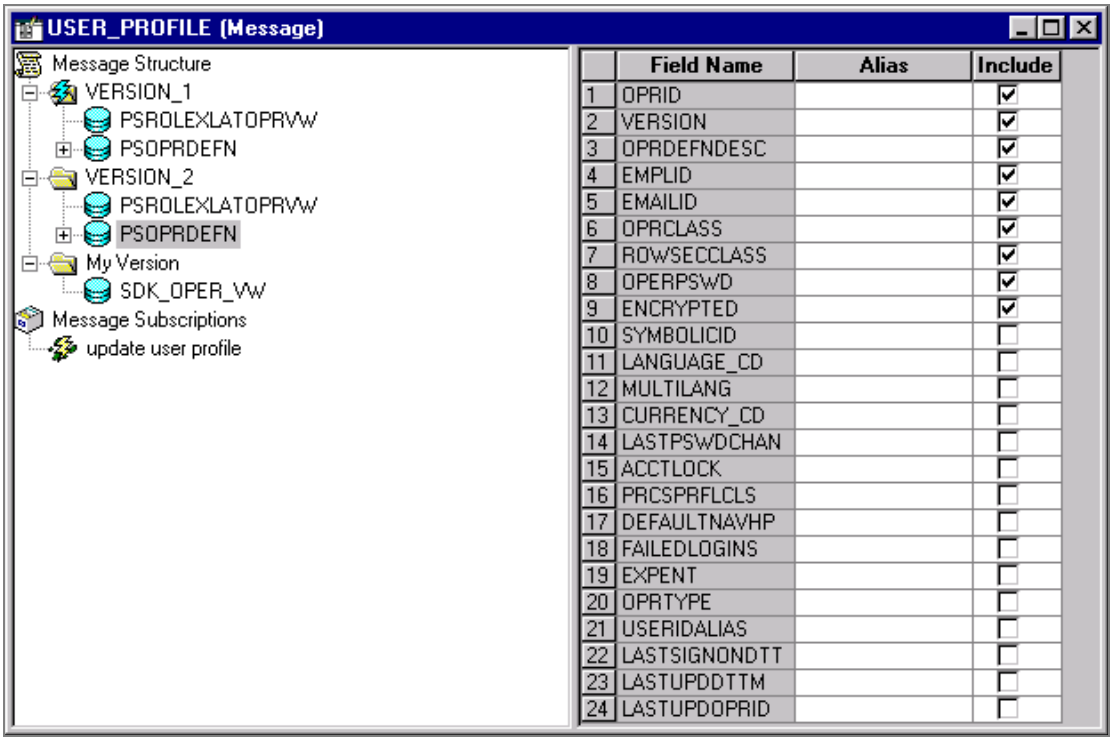
Defining a Message Version

You can insert a new message version or message subscription into the message definition, or insert a record anywhere in the hierarchy of an existing version.

This section explains how to:

- Insert a new version in your message.
- Rename a message version.
- Set the default version.

Inserting a New Version



Message definition containing three versions

To create a new message version:

1. Open an existing message definition.
2. With the message definition open, select Insert, Version.

   You can also right-click anywhere in the message definition and select Insert Version from the pop-up menu. A new version entry appears on the Message Structure tree. Note that a new version *is not* added to the Message Subscriptions section. Your Subscription or OnRequest PeopleCode must select the message version you want to process.

See Also

Accessing Message PeopleCode

Renaming a Version

Each message version has a default name of VERSION_*N*. This naming scheme starts with VERSION_1 and increases in increments of one for each version you insert. You can keep the default names, or provide names more appropriate to your purposes.

To rename a message version, click the message name twice (don't double-click) and type over the existing name. You can also right-click on the message version name and select Rename from the pop-up menu.

Setting the Default Version

The message version marked with the lightning bolt is the default message version. To change which version is the default, right-click on the message version you want as the default, and select Set As Default from the pop-up menu.

You can also set the default version in the message properties dialog. Select the Use tab, and then select the Default Version from the list of available versions.

See Also

Configuring Message Properties

Using Records In a Message Definition

This section explains how to:

- Insert a child record.
- Insert a sibling record.
- Reorganize the record structure.

Understanding Message Record Structure

Important! If your message will handle PeopleSoft record data, you must insert records in the message definition, in an appropriate hierarchical structure. However, if the message data doesn't map to any record hierarchy, don't insert any records at all. You'll supply or receive the data — and its structure — from an external source, using the PeopleCode XmlDoc or SoapDoc classes. See "Sending and Receiving Messages."

You can insert a new record in your message when a version name or a record name is selected in the message structure view. If a version name is selected, you can only insert a record immediately below it in the hierarchy — a *child record.* If a record name is selected, you can also insert a *sibling record* — that is, one at the same level as the selected record.

Note. There's no real difference between a child record and a sibling record, except their relative positions when they're inserted. You can move any record to any location in the version hierarchy after it's been inserted.

Modifying Underlying Record Definitions

Records you insert in a message definition are proxies for the original record definitions. Any change you make to an underlying record definition is automatically reflected by a change in the corresponding record in the message definition.

Fields Defined as Uppercase

If you have a message definition that includes character fields defined as uppercase, when the message is received, character data in those fields is automatically converted to uppercase. This happens when your receiving PeopleCode inserts the message data in a rowset, and overrides any previous changes in the data, including transformation and data translation.

Inserting a Child Record

You must insert the first record into a message version as a child record.

To insert a child record into a message version:

1. In the message structure tree, select the message version or existing record below which you want to insert the new record.
2. Select **Insert, Child Record**.

   You can also right-click the selected message version or record name and select **Insert Child Record** from the pop-up menu.
3. Use the Insert Record dialog to search for, select and insert your desired record.

Note. The Insert Record dialog remains active until you click **Close**. This enables you to insert multiple records quickly and efficiently at the same level as the first one, without having to return to the menu for each record to be inserted.

Inserting a Sibling Record

To insert a sibling record into a message version:

1. In the message structure tree, select an existing record at the same level that you want to insert the new record.
2. Select **Insert, Record**.

   You can also right-click the selected record name and select **Insert Record** from the pop-up menu.
3. Use the Insert Record dialog to search for, select and insert your desired record.

Note. The Insert Record dialog remains active until you click **Close**. This enables you to insert multiple records quickly and efficiently at the same level as the first one, without having to return to the menu for each record to be inserted.

Reorganizing the Record Structure

The message definition provides a set of directional buttons in the toolbar, which you can use to reposition any record within the hierarchy of a message version.

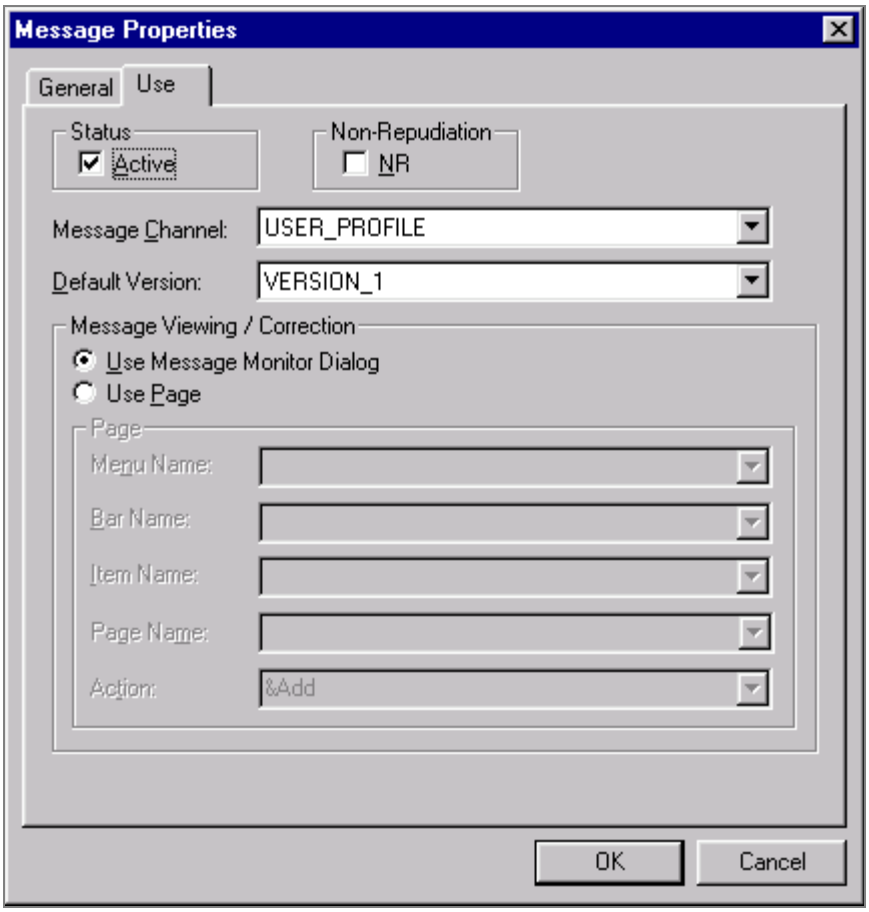
The up and down arrows don't change the level of the selected item, just its order among other items at that level. The left and right arrows move the selected item lower and higher in the message structure hierarchy.

Note. When you reposition a record in the message version structure, its child records are also repositioned, and their child records, and so on.

Configuring Message Properties

The message properties include fields for assigning the message to a channel, specifying which version of the message is the default, and determining the method for viewing and correcting errors in the message.

With a message definition open, click the **Properties** button to access the message properties, and select the Use tab.



Message Properties dialog box — Use tab

Status — Active

Select this check box to indicate that Integration Broker can send this message.

Note. If you clear this check box to deactivate the message, any transactions and relationships that use it are also deactivated. Activating the message won't automatically reactivate the transactions and relationships; they must be individually reactivated. See "Administering Basic Integrations," Editing Transaction Details and "Administering Relationships," Configuring a Relationship for details.

Nonrepudiation — NR

Select this check box to indicate that this message must be sent and received in nonrepudiated form.

Note. You must also configure the nodes that send and receive this message so they support nonrepudiation. See Understanding Nonrepudiation.

Message Channel

Select the message channel to which you want to assign this message. You can't save the message definition without specifying a message channel.

Default Version	Select the version of this message that will be sent or received by default when the sending or receiving PeopleCode doesn't specify a version.
Message Viewing/Correction	Specify the message viewing/correction method you want to use for this message at runtime. Select from the following methods: *Use Message Monitor Dialog:* Use Integration Broker Monitor to view and correct your message. This is the default value. *Use Page:* Use a page of your own design to view and correct your message. See "Sending and Receiving Messages," Processing Inbound Errors for details.

Understanding Nonrepudiation

Integration Broker applies nonrepudiation (NR) capability to cross-node messaging by digitally signing request messages and their replies. NR ensures that none of the nodes in an integration can disavow their participation in a given transaction.

In PeopleSoft applications, NR provides two-way protection — both the request and the reply message are nonrepudiated. Integration Broker uses public key infrastructure technology to implement NR for messaging. Each participating node's keystore contains its own private key, and the public keys of the nodes with which it exchanges messages.

A High Level Description of Nonrepudiation

1. Node A generates a number known as a *digest* that uniquely identifies a request message.
2. Node A encrypts the digest with its private key, and inserts the resulting *signature* into the NR request message.
3. Node A sends the NR request message to node B.
4. When the NR request message is received, node B authenticates the signature in the message using node A's public key in its keystore.
5. Node B generates a digest that uniquely identifies a reply message.
6. Node B encrypts the digest with its private key, and inserts the resulting signature into the NR reply message to confirm receipt of the NR request message.
7. Node B sends the NR reply message to node A.
8. When the NR reply message is received, node A authenticates the signature in the message using node B's public key in its keystore.

NR produces the following result:

- The sending node cannot repudiate that the message was sent, because the receiving node has a copy of the request signed by the sender.

- The receiving node cannot repudiate that the message was received and processed, because the sending node has a copy of the reply signed by the receiver.
- The message integrity is verified, because the validated signature of each NR message assures that the message content as received exactly matches the content as sent.

To save NR messages for future reference, you must make sure they're archived. See "Defining Message Channels," Configuring the Message Channel and "Using Integration Broker Monitor."

Other Considerations

You select a check box in a node definition to indicate that it supports NR, and select a check box in a message definition to indicate that it's an NR message. See "Administering Basic Integrations," Specifying General Node Information.

You must also supply the digital certificates containing the public and private keys required for the transaction. These elements are required at every node that participates in the transaction; Integration Broker handles the mechanics. See *PeopleTools PeopleBook: Security,* "Setting up Digital Certificates and Single Signon," Working With Digital Certificates.

If a participating node doesn't use Integration Broker, that node is still responsible for managing the appropriate public and private keys, inserting properly formatted signatures in the NR messages it sends, and properly handling signatures in messages it receives. See "Applying Transformation, Translation and Filtering," A Message Example.

See Also

"Defining Message Channels"

Accessing Message PeopleCode

This section explains how to:

- Create a message subscription.
- Modify subscription PeopleCode.
- Access and modify OnRequest PeopleCode.
- Generate a test message.

Understanding Message PeopleCode

PeopleCode for sending messages can be located in a number of places, including PeopleCode events associated with records, record fields, components, and Application Engine programs. However, the PeopleCode you use to receive a message must be associated with the message definition. The transmission type of the message determines the location of your program:

- To receive a message asynchronously, create an entry in the Message Subscriptions section of the message structure view, and enter your PeopleCode into its associated *Subscription* PeopleCode event.

- To receive a message synchronously, enter your PeopleCode into the existing *OnRequest* PeopleCode event associated with the root of the message structure.

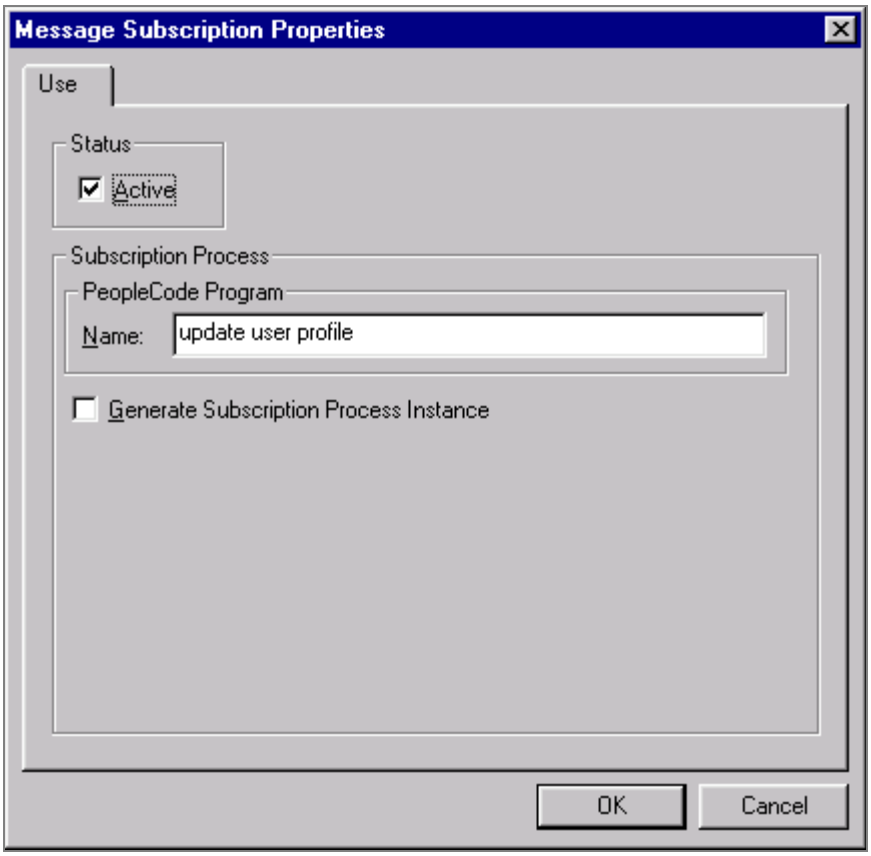
This section describes only how to access PeopleCode associated with a message definition. See "Sending and Receiving Messages" for details about developing the PeopleCode programs for generating, sending, receiving and processing messages.

Creating a Message Subscription

The Subscription event provides the timing for executing a message subscription PeopleCode program. It's associated with a message definition for the purpose of receiving the message asynchronously. Use this PeopleCode to validate and process the message data.

You can create multiple message subscriptions for a given message, each with a different name, inserted in the Message Subscriptions section. Each subscription runs independently in parallel with the others, processing its own copy of the message. In this way, a single inbound message can simultaneously be processed for different purposes in your application. Rather than complicating existing subscription code, you can create a completely separate subscription for each situation.

Because the only purpose of a message subscription object is to represent a PeopleCode program, it necessarily has an associated Subscription event, which contains the PeopleCode. The Subscription event doesn't exist without a message subscription.



Message Subscription Properties dialog box

To create a message subscription:

1. Open a message definition in Application Designer.
2. Right click anywhere in the message structure view, and select **Insert Message Subscription.**

   The Message Subscription Properties dialog box appears.
3. Select the Active check box to enable the subscription.

   If you clear the check box, the subscription won't execute when an instance of this message arrives.
4. Enter a descriptive name for the subscription program.
5. Select the Generate Subscription Process Instance check box to automatically produce a unique number that identifies the process request each time Process Scheduler runs this subscription in batch mode.
6. Click **OK.**

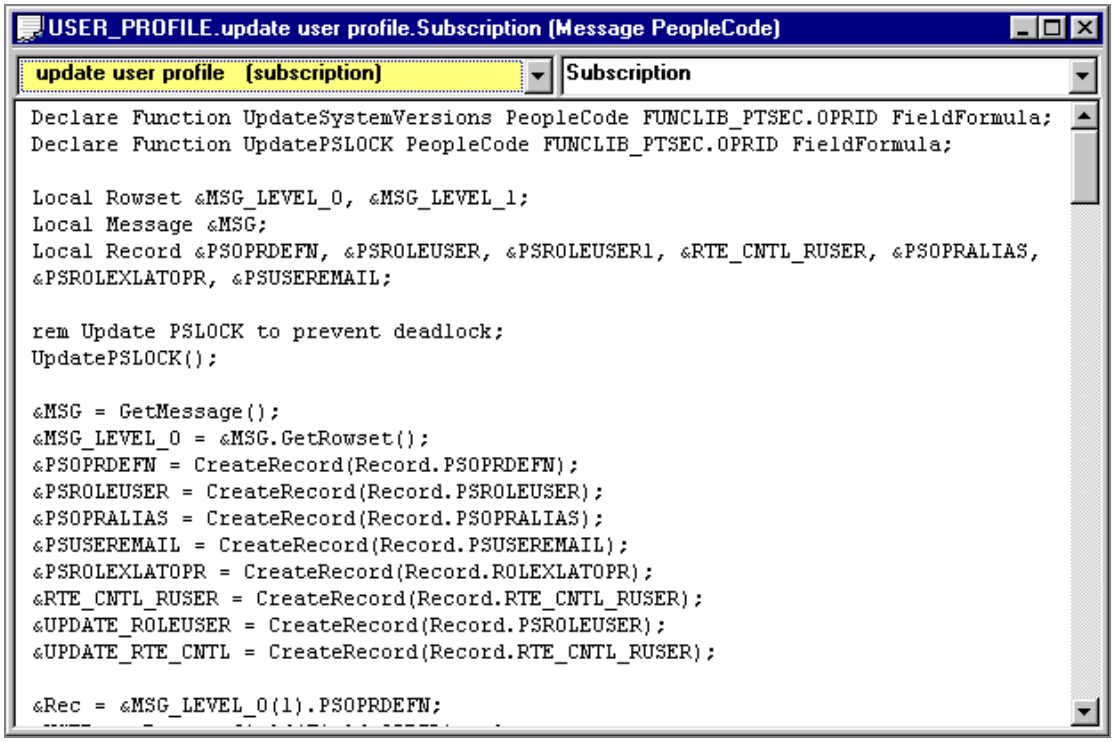
   The Subscription PeopleCode Editor for your subscription appears, and you can proceed to write your asynchronous subscription program.

Note. You can modify the subscription properties at any time. Right click on the subscription name in the Message Structure view, and select Message Subscription Properties to open the Message Subscription Properties dialog box.

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Using the PeopleCode Editor"

Modifying Subscription PeopleCode



Subscription PeopleCode program

To modify Subscription PeopleCode:

1. Open a message definition in Application Designer.
2. Double click the name of the message subscription you want to modify.

   The Subscription PeopleCode Editor for your selected subscription appears, with the subscription program displayed. You can modify the program in the normal way.

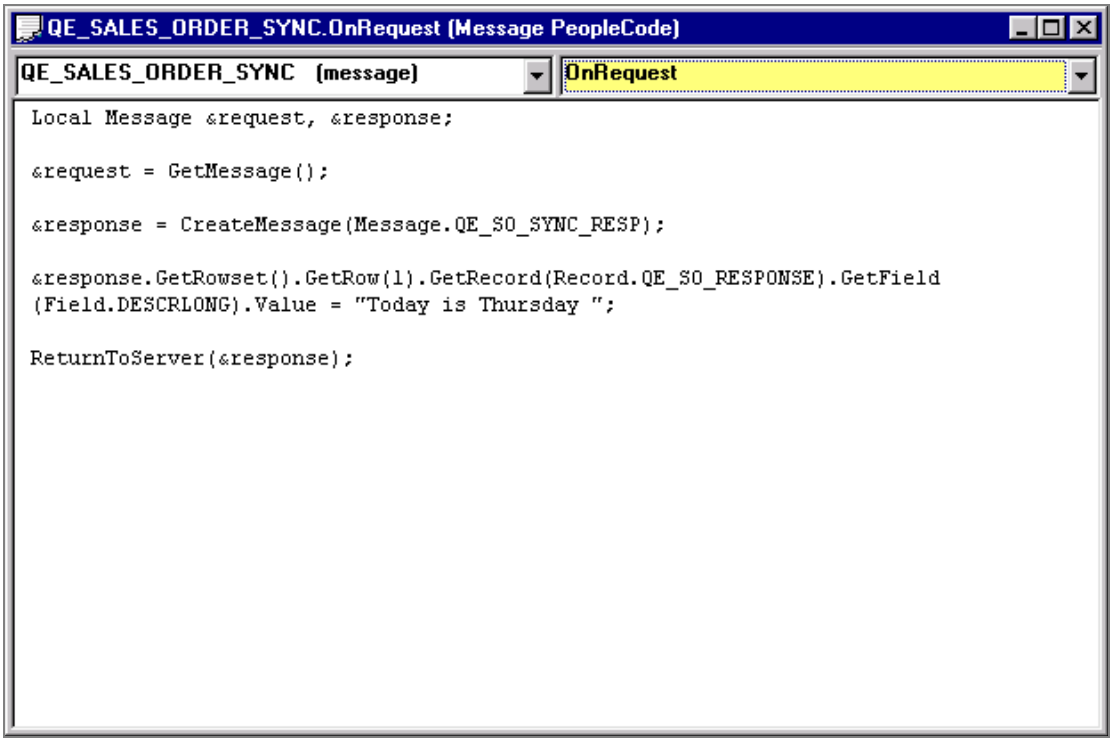
See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Using the PeopleCode Editor"

Accessing and Modifying OnRequest PeopleCode

The OnRequest event provides the timing for executing a PeopleCode program associated with a message definition for the purpose of receiving the request message synchronously. Your program must also generate and populate a synchronous response message. Use this PeopleCode to validate and process the message data.

Unlike message subscriptions, there's only one OnRequest event associated with the root of the message structure, and therefore only one PeopleCode program.



OnRequest PeopleCode program

To access the message OnRequest program:

1. Right click anywhere in the Message Structure section of the message structure view, and select **View PeopleCode.**

   The PeopleCode Editor for your message definition appears.

2. Make sure the message name appears in the left dropdown list, and select **OnRequest** from the right dropdown list.

   The PeopleCode Editor for the OnRequest event appears. You can enter and modify your synchronous program code in the normal way.

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Using the PeopleCode Editor"

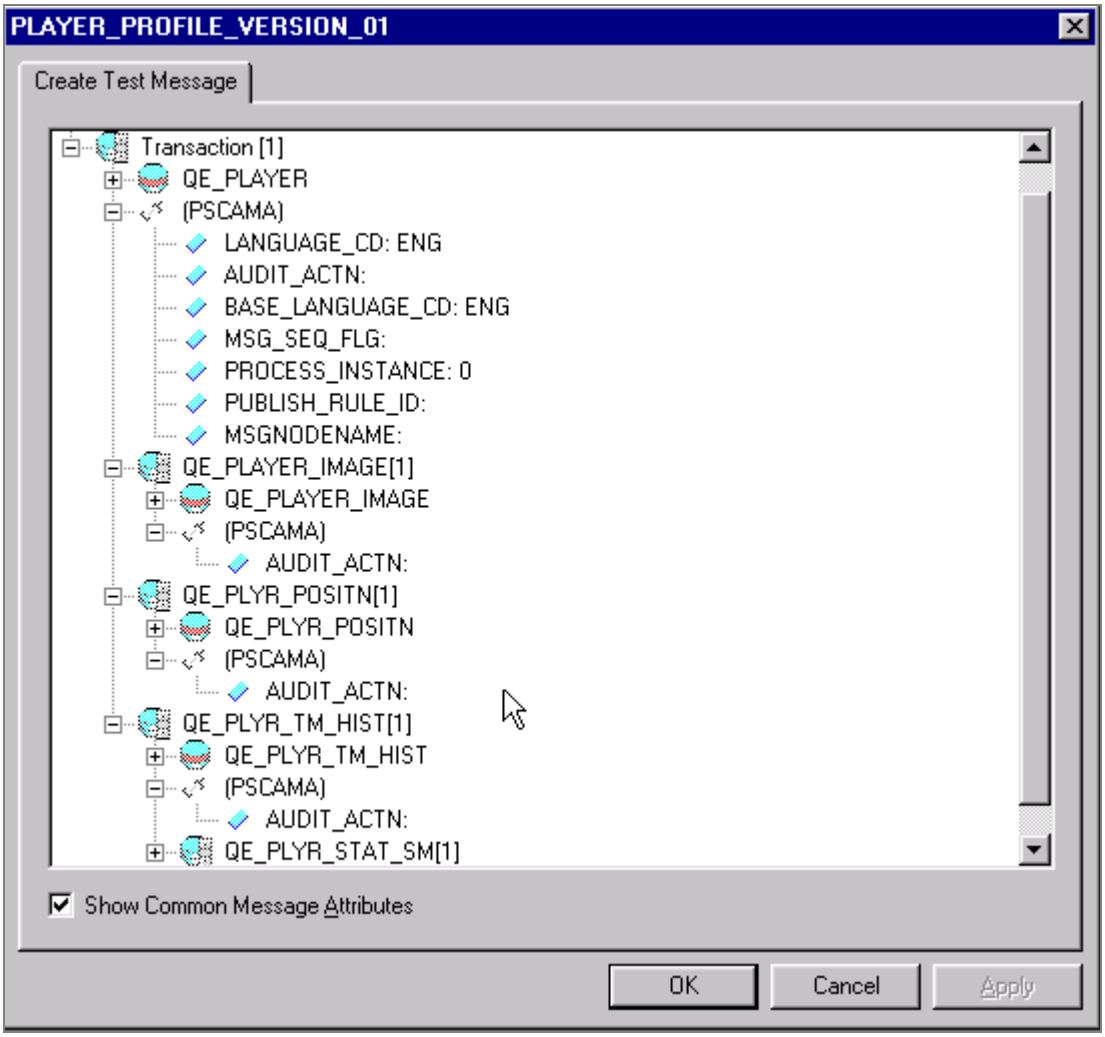
Generating a Test Message

A test message gives you useful information about the selected message definition and its associated PeopleCode. Because the message is not actually sent, creating a test message is not intended to be a *full* send-and-receive test.

To create a test message:

1. In the message definition, right-click the message version you want to test, and select **Create Test Message.**

   A test message appears showing the structure of the message as it would be sent. Select the **Show Common Message Attributes** check box to displays the PSCAMA fields in the test message.



Creating a Test Message

2. Enter sample data for the fields in the message.

You can enter sample data into any field shown in the Create Test Message view. To enter sample data for a field, double-click the field and enter a value. You can also right-click on the field, and select **Edit Value.**

Note. To publish every field in the message, you must enter sample data in at least one field of every record.

---

3. Click **OK** to "publish" the test message.
4. Use Integration Broker Monitor to view the results of the test message.

See Also

"Sending and Receiving Messages," Understanding PSCAMA

"Using Integration Broker Monitor"

CHAPTER 6

Sending and Receiving Messages

This chapter discusses the use of PeopleCode in the messaging process, and explains how to:

- Generate and send a message.
- Receive and process a message.
- Process inbound errors.

Understanding Sending and Receiving Messages

You should already have defined your message channel and message in Application Designer. See "Defining Message Channels" and "Defining Messages."

This chapter discusses creating the basic PeopleCode to send and receive messages. Once your PeopleCode is created, you must also define associated nodes and transactions to implement a complete integration. See "Administering Basic Integrations."

Message Structure Considerations

Integration Broker is delivered with accompanying PeopleCode functions and classes to build and access message content structured according to three different protocols:

- PeopleSoft standard rowset based messages. See PeopleSoft Rowsets.
- XML Document Object Model compliant messages. See XML Document Object Model.
- Simple Object Access Protocol messages. See Simple Object Access Protocol.

PeopleSoft Rowsets

To work with rowset based messages — the PeopleSoft native format — you define a message in Application Designer by inserting records into the message definition in a hierarchical structure, then populate the message and manipulate its contents using the PeopleCode Rowset and Message classes. Externally, the message is transmitted as XML with a prescribed PeopleSoft schema.

The PeopleSoft message schema includes a record called PSCAMA (PeopleSoft Common Application Message Attributes), which PeopleTools adds to every level of the message

structure to convey basic information about the message and its data rows. See Understanding PSCAMA.

When to Use the Rowset and Message Classes

Using the Rowset and Message classes is the recommended approach for messaging between PeopleSoft applications. If a message is populated with data from a PeopleSoft application's database or component buffer, the Message class is best suited for handling that data.

This chapter includes some examples of PeopleCode for generating, sending, receiving and processing messages using the Rowset and Message classes. You'll find more comprehensive coverage of those classes in the PeopleCode Reference Guide.

See Also

"Defining Messages," Accessing Message PeopleCode

PeopleTools PeopleBook: PeopleCode Reference, "Rowset Class"

PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

XML Document Object Model

The World Wide Web Consortium (W3C) has established a document object model (DOM) for accessing and manipulating structured data. A DOM is a specification of how data should be presented for access using programming languages, regardless of the data's actual internal representation. The DOM specifies a standardized application programming interface (API) that provides a consistent, familiar way to work with almost any data. This API — the XML DOM — enables you to easily create, retrieve, navigate and modify messages.

You define a message in Application Designer without inserting any records to define its structure, then populate the message and manipulate its contents using the PeopleCode XmlDoc class and built-in functions, which comply with the XML DOM. If a message is populated with XML data from an external file or URL, the XmlDoc class is well suited for handling that data.

Note. To use XmlDoc you must create a message definition without inserting any records. See "Defining Messages." You can use the XmlDoc class to access inbound rowset based messages, because they're XML DOM compliant; however, the Message and Rowset classes handle the PeopleSoft native format more easily.

When to Use the XmlDoc Class

Use the XmlDoc PeopleCode class if any of the following is true:

- Your message structure doesn't fit the PeopleSoft rowset model.
- Your message data doesn't come from PeopleSoft database records.

- Your third party source or target node requires non-XML messages.

Although you can use the XmlDoc PeopleCode class to generate or process messages that use the SOAP protocol, it's bound to be a cumbersome process; the SoapDoc class is strongly recommended for this.

Note. The XmlDoc class does play a part in sending and receiving SOAP messages. See Simple Object Access Protocol.

This chapter includes some examples of PeopleCode for generating, sending, receiving and processing messages using the XmlDoc class. You'll find more comprehensive coverage of XmlDoc in the PeopleCode PeopleBooks.

See Also

Generating and Sending a Message

Receiving and Processing a Message

"Applying Transformation, Translation and Filtering"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

Simple Object Access Protocol

The W3C has established a specification for using Simple Object Access Protocol (SOAP) for defining synchronous transactions in a distributed Web environment. SOAP is appropriate for UDDI interactions, or to interact with SOAP-compliant servers.

You define a message in Application Designer without inserting any records to define its structure, then populate the message and manipulate its contents using the PeopleCode SoapDoc class and built-in functions, which comply with the W3C SOAP specification. If a message is populated with SOAP compliant XML data, the SoapDoc class is well suited for handling that data.

The XmlDoc Connection

Like XmlDoc, SoapDoc complies with the W3C XML DOM specification. The SoapDoc class is actually based on the XmlDoc class, with some identical methods and properties. To send and receive SoapDoc messages, you must convert them to XmlDoc objects, and use the XMLDoc functions SyncRequestXmlDoc and GetMessageXmlDoc, respectively. SoapDoc provides a property for handling the conversion easily.

Note. You can use the XmlDoc class to access inbound SOAP based messages, because they're XML DOM compliant; however, the SoapDoc class handles the SOAP format more easily.

When to Use the SoapDoc Class

Use the SoapDoc PeopleCode class if all of the following are true:

- Your third party source or target node requires SOAP-compliant messages.
- Your third party source or target node requires synchronous transactions.
- Your message conforms to the SOAP specification.

This chapter includes some examples of PeopleCode for generating, sending, receiving and processing messages using the SoapDoc and XmlDoc classes. You'll find more comprehensive coverage of SOAP in the PeopleCode PeopleBooks.

See Also

Generating and Sending a Message

Receiving and Processing a Message

PeopleTools PeopleBook: PeopleCode Reference, "SOAPDoc Class"

Transaction Process Flow

Integration Broker supports four types of messaging transactions: Outbound transactions can send messages either asynchronously or synchronously, and inbound transactions can receive messages either asynchronously or synchronously. Each node participating in an integration must apply at least one transaction. See "Administering Basic Integrations," Configuring Transactions.

Each transaction type requires a sequence of actions by the participants, as follows:

Transaction Type	*Actions*
Outbound Asynchronous	1. Your application generates and sends a request message. 2. One or more target nodes receive and process the request message.
Outbound Synchronous	1. Your application generates and sends a request message. 2. Your application suspends activity and waits for a response message. 3. A single target node receives and processes the request message, then generates and sends a response message. 4. Your application resumes its activity, receives and processes the response message.
Inbound Asynchronous	1. A source node generates and sends a request message. 2. Your application receives and processes the request message.

Transaction Type	*Actions*
Inbound Synchronous	1. A source node generates and sends a request message. 2. The source node suspends activity and waits for a response message. 3. Your application receives and processes the request message, then generates and sends a response message. 4. The source node resumes its activity, receives and processes the response message.

For any transaction type, your application must invoke PeopleCode to generate and send a message, or to receive and process a message.

Note. When Integration Broker sends a message, the receiving system sends a low level **reply** message back to the sender. This differs from the **response** message, which is the second half of a synchronous transaction. A reply message serves only a "handshaking" purpose, to notify the sending system of the transmission status of the request or response message. It's processed automatically by the application server, which uses that status information to update Integration Broker Monitor. See "Using Integration Broker Monitor."

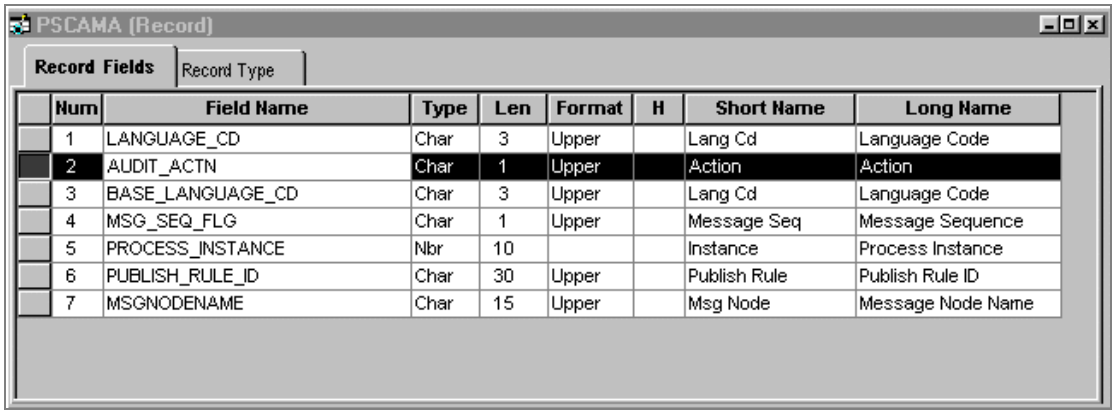
Understanding PSCAMA

PSCAMA, an acronym for *PeopleSoft Common Application Message Attributes,* is the name of a record that PeopleTools adds to every level of the message structure during processing. It isn't accessible in the message definition, but you can reference it as part of the message object in the sending and receiving PeopleCode, and see it in the Integration Broker Monitor. PeopleCode processes this record the same way as any other record.

Note. PSCAMA records are automatically included in your messages only if you insert database records to define the message structure. You can use the XmlDoc class to handle an inbound message containing PSCAMA records, but the Message class is much better suited for this. PSCAMA isn't compatible with the SOAP format.

PSCAMA contains fields common to all messages. The <PSCAMA> tag repeats for each row in each level of the <Transaction> section of the message. The sender can set the PSCAMA fields to give the recipient basic information about the message, for example to indicate the message language or the type of transaction a row represents. When receiving a message, your PeopleCode should inspect the PSCAMA records for this information and respond accordingly.

The PSCAMA Record



Num	Field Name	Type	Len	Format	H	Short Name	Long Name
1	LANGUAGE_CD	Char	3	Upper		Lang Cd	Language Code
2	AUDIT_ACTN	Char	1	Upper		Action	Action
3	BASE_LANGUAGE_CD	Char	3	Upper		Lang Cd	Language Code
4	MSG_SEQ_FLG	Char	1	Upper		Message Seq	Message Sequence
5	PROCESS_INSTANCE	Nbr	10			Instance	Process Instance
6	PUBLISH_RULE_ID	Char	30	Upper		Publish Rule	Publish Rule ID
7	MSGNODENAME	Char	15	Upper		Msg Node	Message Node Name

PSCAMA record definition

PSCAMA Fields

Field Name	*Description*
LANGUAGE_CD	Required. Indicates the language in which the message is generated. When sending with component PeopleCode, the system sets this field to the operator's default language code. See Language Codes.
AUDIT_ACTN	Required. Identifies each row of data as an Add, Change, or Delete action. See Audit Action Codes.
BASE_LANGUAGE_CD	(Optional) Indicates the base language of the sending database. Used by generic full table subscription PeopleCode to help determine which tables to update.
MSG_SEQ_FLG	(Optional) Supports the transmission of large transactions that may span multiple messages. Indicates whether the message is a header (***H***), trailer (***T***) or contains data (blank). The header and trailer messages don't contain any message data. The receiving system can use this information to determine the start and end of the set of messages and initiate processes accordingly. For example, the header message might cause staging tables to be cleared while the trailer might indicate that all the data has been received and an update job should be initiated.
PROCESS_INSTANCE	(Optional) Process instance of the batch job that created the message. Along with the sending node and publication ID, this can be used by the receiving database to uniquely identify a group of messages from the sending node.
PUBLISH_RULE_ID+	(Optional) Indicates which publish rule was invoked to create the message. Used by routing PeopleCode to locate the appropriate chunking rule, which then determines to which nodes the message should be sent.

Field Name	Description
MSGNODENAME+	(Optional) The node to which the message should be sent. This field is passed to the Publish utility by the Application Engine program. Routing PeopleCode must look for a value in this field and return that value to the application server.

+Third party applications can ignore these fields.

Language Codes

Each message can contain only one language code, in the first PSCAMA record. The sending application sets this code (the LANGUAGE_CD field) to indicate the language in which the message is generated, so the receiving application can take that information into account when processing the message.

PeopleSoft language codes contain three characters, and are mapped to corresponding ISO locale codes in an external properties file. This mapping enables the PeopleSoft Internet Architecture to derive certain defaults from the ISO locales that are stored in a user's browser settings. Your PeopleSoft application is delivered with a set of predefined language codes; you can define custom codes as well.

Note: There can be only one language code for the entire message. If you need to send messages in multiple languages, send multiple messages.

See Also

PeopleTools PeopleBook: PeopleSoft Globalization, "Controlling International Preferences"

Audit Action Codes

A PSCAMA record appears following each row of the message. At minimum it contains an audit action code (the AUDIT_ACTN field), denoting the action to be applied to the data row. The audit action is required so the receiving system knows how to process the incoming data.

The valid audit action codes match those used in PeopleSoft's audit trail processing:

A, ***C***, ***D***, ***K***, ***N***, ***O***, blank

The audit action values are set by the system or by your sending PeopleCode to specify that a record should be added, changed, or deleted, as follows:

- To delete a row, the message record's AUDIT_ACTN value is ***D***.
- To add a non-effective dated row, the message record's AUDIT_ACTN value is ***A***.
- To add an effective dated row, the message record's AUDIT_ACTN value is ***A***.

If you populate the row data using the PeopleCode Message class method CopyRowsetDeltaOriginal, an additional record is created with an AUDIT_ACTN value of ***O***, containing the original values of the current effective dated row.

- To change only non-key values in a row, the message record's AUDIT_ACTN value is ***C***.

If you populate the row data using the PeopleCode Message class method CopyRowsetDeltaOriginal, an additional record is created with an AUDIT_ACTN value of ***O***, containing the original values of the current effective dated row.

- To change at least one key value in a row (in addition to any non-key values), the message record's AUDIT_ACTN value is ***N***.

If you populate the row data using the PeopleCode Message class methods CopyRowsetDeltaOriginal or CopyRowsetDelta, an additional record is created with an AUDIT_ACTN value of ***K***, containing the original values of the current effective dated row.

- If a row's content hasn't changed, the message record's AUDIT_ACTN value is blank. This is the default value, which is also used to tag the parents of rows that have changed.

Other PSCAMA Considerations

- You can update values on the PSCAMA record just like any other record in the message definition before sending the message.

- The receiving process can access the fields on this record just like any other fields in the message.

- The size of the PSCAMA record varies—in particular, notice a difference between the first PSCAMA record and the ones that follow. The first record contains all of the fields, while the other PSCAMA records have only the AUDIT_ACTN field. This is because the AUDIT_ACTN field is the only field that can differ for each row of data. Every message will contain the first PSCAMA record with all of the fields.

- Although the first PSCAMA record is always present in a message, not all of the remaining PSCAMA records are sent in the message. If a <PSCAMA> tag is not included for a specific row in a message, then you can assume the row has not changed. Therefore, the PSCAMA tag should be present on any row that has changed. In addition, if the PSCAMA <AUDIT_ACTN> tag is blank (***<AUDIT_ACTN>" "</AUDIT_ACTN>***) or null (***<AUDIT_ACTN></ AUDIT_ACTN>*** or ***<AUDIT_ACTN />***), you can also assume the row hasn't changed.

- If you're receiving a message that has incremental changes, only the rows that have changed *and their parent rows* are present in the message.

You can view an example of an outbound message with the PSCAMA records inserted by testing your message definition. See "Defining Messages," Generating a Test Message.

Generating and Sending a Message

This section describes how to:

- Handle outbound asynchronous transactions.
- Handle outbound synchronous transactions.
- Handle cookies in messages.

Understanding Outbound Messaging

Message Order

Integration Broker guarantees that messages are delivered in the order sent and that they are single-threaded at the PeopleSoft receiving node. However, message order is not part of the channel definition. Therefore, it's your responsibility to send messages in the proper order.

For example, you can't specify in the channel definition that a message named PO_CREATE be processed before a message named PO_UPDATE. If you send PO_UPDATE before PO_CREATE, the recipient processes PO_UPDATE first.

Note. You can modify this behavior using channel partitioning. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Testing Your Message

Make sure that you adequately unit test and system test your messages. You can easily unit test your message by triggering the PeopleCode that sends the message, and then viewing it in the Integration Broker Monitor. From the Integration Broker Monitor, you can view the contents of each field to verify that the message data is formatted correctly. See "Using Integration Broker Monitor."

You can also trigger a message with the Test Message feature in Application Designer. See "Defining Messages," Generating a Test Message.

Tips for Message Class Outbound PeopleCode

- Use SelectByKey whenever possible to get data that isn't in the component buffer.
- Shortcut — if your record names are the same, use the standard record methods, such as SelectByKey, Insert, Update, on the message records.
- Use "If &Msg.Size > %MaxMessageSize" at Level 0 to control message size when dealing with multiple transactions.

Note. The **OnRouteSend** event associated with the message definition enables you to apply PeopleCode that filters the destination nodes to which Integration Broker routes the message. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Handling Non-XML Data

If you're generating a non-XML outbound message in your PeopleSoft application, it's up to you to insert your message content into a special *xml* section containing a *CDATA* tag, as follows:

```
<xml psnonxml="yes">
   <![CDATA[nonXML_message_data]]>
</xml>
```

See Also

PeopleTools PeopleBook: PeopleCode Reference, "Record Class"

Handling Outbound Asynchronous Transactions

Use the PeopleCode Message class Publish method or the PublishXmlDoc built-in function. You can employ Publish or PublishXmlDoc in any of several places to asynchronously send your message:

- A record field PeopleCode event
- A component PeopleCode event
- An Application Engine program
- A message subscription program
- The message OnRequest PeopleCode event

Note. Message subscriptions and the OnRequest event are triggered only when an inbound transaction occurs, but the receiving PeopleCode can also send messages.

Message Class Outbound Asynchronous Example

The following example uses the Message class Publish method. See *PeopleTools PeopleBook: PeopleCode Reference,* "Message Class" for more information.

```
Local Message &MSG;
Local Rowset &SALES_ORDER, &RS;

/*Get a pointer to the component buffer rowset */
&SALES_ORDER = GetLevel0();
```

```
/*Create an instance of the SALES_ORDER_ASYNC message object */
&MSG = CreateMessage(Message.SALES_ORDER_ASYNC);

/*Copy the rows from the rowset to the message object */
&MSG.CopyRowset(&SALES_ORDER);
/*Send the message */
&MSG.Publish();
```

XmlDoc Class Outbound Asynchronous Example

The following example uses the XmlDoc class PublishXmlDoc built-in function. See *PeopleTools PeopleBook: PeopleCode Reference,* "XmlDoc Class" for more information.

```
Local XmlDoc &xmlRequestDoc;

/* Create an XmlDoc object */
&MyDoc = CreateXmlDoc();

/* Parse an input XML file into an XmlDoc */
&xmlRequestDoc = ParseXmlFromURL("C:\pt\appserv\files\input.xml");

/* Send request */
PublishXmlDoc(&xmlRequestDoc, Message.MyMessage, Node.MyNode);
```

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Understanding PeopleCode and Events"

PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

Handling Outbound Synchronous Transactions

Note. The response message returned from an outbound synchronous transaction is no different than an inbound request message. Once you have it in a Message, XmlDoc or SoapDoc object, it has the same properties — and can be treated exactly the same way — as any other object of that type. See Receiving and Processing a Message for details.

Use the PeopleCode Message class SyncRequest method or the SyncRequestXmlDoc built-in function. You can employ SyncRequest or SyncRequestXmlDoc in one of the following PeopleCode events to synchronously send your message:

- The record field SavePreChange PeopleCode event
- The record field SavePostChange PeopleCode event

- The record field Workflow PeopleCode event
- The record field FieldChange PeopleCode event
- The message OnRequest PeopleCode event
- A message subscription program

Note. The OnRequest event and subscription programs are triggered only when an inbound transaction occurs, but when the receiving PeopleCode executes, it can also send messages.

Message Class Outbound Synchronous Example

The following example uses the message class SyncRequest method.

```
Local Message &MSG, &response;
Local Rowset &SALES_ORDER;

&SALES_ORDER = GetLevel0();
&MSG = CreateMessage(Message.SALES_ORDER_SYNC);
&MSG.CopyRowsetDelta(&SALES_ORDER);

/* send the synchronous request; the return value is the response message
object */
&response = &MSG.SyncRequest();

/* check the response status; 0 means OK */
If (&response.ResponseStatus = 0) Then
     /* process the response */
     MY_SALES_ORDER_SYNC.ORDER_ID =
     &response.GetRowset().GetRow(1).GetRecord(Record.SO_RESPONSE).GetField(Fie
     ld.ORDER_ID).Value);
else

     /* do error handling */

End-If;
```

XmlDoc Class Outbound Synchronous Example

The following example uses the SyncRequestXmlDoc built-in Function.

```
Local string &xmlString;
Local XmlDoc &xmlRequestDoc;
Local XmlDoc &xmlResponseDoc;
Local Field &DescrLong;

&DescrLong = GetLevel0().GetRow(1).GetRecord(Record.QE_FUNCLIB_IB).DESCRLONG;
```

```
/* Get the input XML string */
&xmlString = &DescrLong.Value;

/* Parse the input XML string into an XmlDoc */
&xmlRequestDoc = CreateXmlDoc(&xmlString);

/* Send request */
&xmlResponseDoc = SyncRequestXmlDoc(&xmlRequestDoc, Message.XMLSYNCREQ);

/* Display the output XML string */
&DescrLong.Value = &xmlResponseDoc.GenXmlString();
```

SoapDoc Class Outbound Synchronous Example

The following example uses the SyncRequestXmlDoc built-in Function.

Note. Because SyncRequestXmlDoc is an XmlDoc function, you must convert your SoapDoc request message to an XmlDoc object before sending it. Likewise, the response message is an XmlDoc object, so you must convert it to a SoapDoc object to process it using SOAP methods.

```
Local File &MY_FILE;

Local XmlDoc &request, &response;
Local string &strXml;
Local SoapDoc &soapReq, &soapRes;

&soapReq = CreateSoapDoc();
&bool = &soapReq.ParseXmlFromURL("C:\pt\appserv\files\inputSoap.xml");

&strXml = &soapReq.GenXmlString();
&MY_FILE = GetFile("c:\temp\syncInput.txt", "w", "a", %FilePath_Absolute);
&MY_FILE.WriteLine("YEAH!");
&MY_FILE.WriteLine(&strXml);
&MY_FILE.Close();

/* Convert request from SoapDoc to XmlDoc before sending */
&request = &soapReq.XmlDoc;
&response = SyncRequestXmlDoc(&request, Message.QE_SOAP_REQ, Node.UNDERDOG);

&soapRes = CreateSoapDoc();
/* Convert response from XmlDoc back to SoapDoc */
&soapRes.XmlDoc = &response;
&OK = &soapRes.ValidateSoapDoc();
&strXml = &soapRes.GenXmlString();
```

```
&MY_FILE = GetFile("c:\temp\sync.txt", "w", "a", %FilePath_Absolute);
&MY_FILE.WriteLine(&strXml);
&MY_FILE.Close();
```

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Understanding PeopleCode and Events"

PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

PeopleTools PeopleBook: PeopleCode Reference, "SOAPDoc Class"

Handling Cookies

Integration Broker provides basic cookie handling for exchanges initiated by your PeopleSoft application. You can accept a synchronous response message containing cookies, save those cookies in a global variable, and later return them to the target node in an outbound synchronous or asynchronous request message. This is a typical application of cookies in a Web interaction.

Cookies are implemented as a property, *Cookies*, of a Message object. You can access this property only in an inbound synchronous response message or an outbound request message.

Receiving Cookies

This example retains the cookies from a response message to a global variable:

```
Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(Message.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* send the synchronous request; the return value is the response message
object */
&SalesResponse = &SalesRequest.SyncRequest();

/* Retrieve cookies from the response message */
&SalesCookies = &SalesResponse.Cookies;
```

Returning Cookies

This example retrieves the previously retained cookies from the global variable and inserts them into a new request message:

```
Local Message &SalesRequest, &SalesResponse;
Local Rowset &SALES_ORDER;
Global string &SalesCookies;

&SALES_ORDER = GetLevel0();
&SalesRequest = CreateMessage(Message.SALES_ORDER_SYNC);
&SalesRequest.CopyRowsetDelta(&SALES_ORDER);

/* Insert the cookies in the request message */
&SalesRequest.Cookies = &SalesCookies;

/* Send the asynchronous request */
&SalesRequest.Publish();
```

Receiving and Processing a Message

This section explains how use PeopleCode to:

- Handle inbound asynchronous transactions.
- Handle inbound synchronous transactions.

Note. The **OnRouteReceive** event associated with the message definition enables you to apply PeopleCode that determines whether the default local node accepts the inbound message. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Handling Inbound Asynchronous Transactions

Use the PeopleCode GetMessage or GetMessageXmlDoc built-in function, and insert your PeopleCode in a message subscription.

Message Class Inbound Asynchronous Example #1

The following basic example uses the GetMessage built-in function.

```
Local Message &MSG;
Local Rowset &LEVEL0;
Local Record &SALES_ORDER_INFO, &REC;

&MSG = GetMessage();
&SALES_ORDER_INFO = CreateRecord(Record.QE_SALES_ORDER);

&LEVEL0 = &MSG.GetRowset();
&REC = &LEVEL0(1).QE_SALES_ORDER;
```

```
&REC.CopyFieldsTo(&SALES_ORDER_INFO);

SQLExec("DELETE FROM PS_QE_SALES_ORDER");

&SALES_ORDER_INFO.Insert();
```

Message Class Inbound Asynchronous Example #2

The following example shows subscription PeopleCode that checks the PSCAMA to determine what the Audit Action code is. It also checks the language code to determine if related language processing is needed.

```
/* Simple PeopleCode Subscribe- - Check the PSCAMA*/
Local Message &MSG;
Local Rowset &MSG_RS;
Local Record &REC_NAME_PREFIX, &REC, &REC_RL;

&MSG = GetMessage();
&MSG_RS = &MSG.GetRowset();
For &I = 1 To &MSG_RS.RowCount
   /* Loop through all transactions */
   &REC = &MSG_RS.GetRow(&I).GetRecord(Record.NAME_PREFIX_TBL);

   /* Get Audit Action for this transaction */
   &ACTION = &MSG_RS.GetRow(&I).PSCAMA.AUDIT_ACTN.Value;
   /* Get Language code for this transaction */
   &LNG = &MSG_RS.GetRow(&I).PSCAMA.LANGUAGE_CD.Value;
   &BASE_LNG = %Language;

   Evaluate &ACTION
      /*****************************/
      /******** Add a Record *******/
      /*****************************/
   When "A"
      /* Add the base language record */
      &REC_NAME_PREFIX = CreateRecord(Record.NAME_PREFIX_TBL);
      &REC.CopyFieldsTo(&REC_NAME_PREFIX);
      &REC_NAME_PREFIX.ExecuteEdits();
      If &REC_NAME_PREFIX.IsEditError Then
         /* error handling */
      Else
         &REC_NAME_PREFIX.Insert();
         /* Need error handling here if insert fails */
         If &LNG <> %Language Then
            /* add related language record */
            &RELLNG = &REC.RelLangRecName;
            &REC_RL = CreateRecord(Record.NAME_PREFIX_LNG);
```

```
            &REC.CopyFieldsTo(&REC_RL);
            &REC_RL.LANGUAGE_CD.Value = &LNG;
            &REC_RL.Insert();
            /* Need error handling here if insert fails */
         End-If;
      End-If;

      /****************************/
      /***** Change a Record *******/
      /****************************/
      /**** Using record objects ***/
   When "C"
      /* Get the Record - insert it */
      &KEY1 = &REC.GetField(Field.NAME_PREFIX).Value;
      &REC_NAME_PREFIX = CreateRecord(Record.NAME_PREFIX_TBL);

      &REC_NAME_PREFIX.NAME_PREFIX.Value =
&REC.GetField(Field.NAME_PREFIX).Value;
      If &REC_NAME_PREFIX.SelectByKey() Then

         &REC.CopyFieldsTo(&REC_NAME_PREFIX);
         &REC_NAME_PREFIX.ExecuteEdits();
         If &REC_NAME_PREFIX.IsEditError Then
            /* error handling */
         Else
            &REC_NAME_PREFIX.Update();
         End-If;
      Else
         &REC.CopyFieldsTo(&REC_NAME_PREFIX);
         &REC_NAME_PREFIX.ExecuteEdits();
         If &REC_NAME_PREFIX.IsEditError Then
            /* error handling */
         Else
            &REC_NAME_PREFIX.Insert();
         End-If;
      End-If;

      /****************************/
      /****** Delete a Record ******/
      /****************************/
      /*** Using SQLExec ***********/
   When "D"
      /* Get the Record using SQLExec- error */
      &KEY1 = &REC.GetField(Field.NAME_PREFIX).Value;
      SQLExec("Select NAME_PREFIX from PS_NAME_PREFIX_TBL where NAME_PREFIX =
:1", &KEY1, &KEY2);
```

```
      If None(&KEY2) Then
         /* Send to error log */
      Else
         SQLExec("Delete from PS_NAME_PREFIX_TBL where NAME_PREFIX = :1",
&KEY1);
         SQLExec("Delete from PS_NAME_PREFIX_LNG where NAME_PREFIX = :1",
&KEY1);
      End-If;

   End-Evaluate;
End-For;
```

Message Class Inbound Asynchronous Example #3

The following example shows subscription PeopleCode with multiple transactions:

```
Local Message &STOCK_MSG;
Local Rowset &HDR_RS, &LN_RS;
Local Record &HDR_REC, &hdr_rec_msg, &LN_REC, &LN_REC_MSG;

/*
This subscription peoplecode processes messages that may contain multiple
Header (MSR_HDR_INV) transactions that may have multiple line (DEMAND_INF_INV)
transactions. The data is inserted into a identically structured header/line
tables (MSR_HDR_INV2 and DEMAND_INF_INV2)
*/

/* Get the STOCK_MSG App. Message rowset */
&STOCK_MSG = GetMessage();

/* Create record objects for the Header and Lines that will be inserted into
*/
&HDR_REC = CreateRecord(Record.MSR_HDR_INV2);
&LN_REC = CreateRecord(Record.DEMAND_INF_INV2);

/* Create an App. Message Rowset that includes the MSR_HDR_INV (Header) and
DEMAND_INF_INV (Line)*/
&HDR_RS = &STOCK_MSG.GetRowset();

/* Clear out all the Headers and Lines */
SQLExec("DELETE FROM PS_MSR_HDR_INV2 WHERE BUSINESS_UNIT = 'M04A1'");
SQLExec("DELETE FROM PS_DEMAND_INF_INV2 WHERE BUSINESS_UNIT = 'M04A1'");

/* Loop through all the headers in the message */
For &I = 1 To &HDR_RS.ActiveRowCount
   /* Instantiate the row within the Header portion of the
      App Message rowset to which data will be copied */
   &hdr_rec_msg = &HDR_RS.GetRow(&I).GetRecord(Record.MSR_HDR_INV);
```

```
   /* Copy data from the level 0 (Header portion) of &STOCK_MSG message
structure
      into the Header record object */
   &hdr_rec_msg.CopyFieldsTo(&HDR_REC);

   &HDR_REC.Insert();

   /* Create Rowset that includes the DEMAND_INF_INV (Line) portion of the App
Message Rowset */
   &LN_RS = &HDR_RS(&I).GetRowset(1);
   /* Loop through all the lines within this header transaction */
   For &J = 1 To &LN_RS.ActiveRowCount
      /* Instantiate the row within the Line portion of the
         App Message rowset to which data will be copied */
      &LN_REC_MSG = &LN_RS.GetRow(&J).GetRecord(Record.DEMAND_INF_INV);
      /* copy data into the Level 1 (Line portion) of &STOCK_MSG object */
      &LN_REC_MSG.CopyFieldsTo(&LN_REC);
      &LN_REC.Insert();
   End-For;
End-For;
```

Message Class Inbound Asynchronous Example #4

There is practical limit to how large a message can be and this can be controlled by a system-wide variable called **%MaxMessageSize.** The system administrator can change this size in the PSOPTIONS settings. It can not be set for each individual message definition, but rather for all messages.

PeopleCode that populates a message object should include code similar to the following example to check the message size before inserting a new Level 0 row.

Note: Always code your %MaxMessageSize checkpoint at the Level 0 record. A batch of transactions can be split across multiple messages, but do not split a single transaction (Logical Unit of Work) into multiple messages.

```
Local SQL &hdr_sql, &ln_sql;
Local Message &STOCK_MSG;
Local Rowset &hdr_rs, &ln_rs;
Local Record &hdr_rec, &ln_rec, &hdr_rec_msg, &ln_rec_msg;

/* This PeopleCode will publish messages for a simple Header/Line record
combination. Multiple Header/Lines are copied to the message until the
%MaxMessageSize is exceeded at which point a new message is built. This
references MSR_HDR_INV (Header) and DEMAND_INF_INV (Line) records */

/* Create an instance of the STOCK_REQUEST message */
&STOCK_MSG = CreateMessage(Message.STOCK_REQUEST);
```

```
/* Create an App. Message Rowset that includes the MSR_HDR_INV (Header) and
DEMAND_INF_INV (Line)*/
&hdr_rs = &STOCK_MSG.GetRowset();

/* Create a SQL object to select the Header rows */
&hdr_sql = CreateSQL("Select * from PS_MSR_HDR_INV WHERE BUSINESS_UNIT='M04A1'
AND ORDER_NO LIKE 'Z%' ORDER BY BUSINESS_UNIT,ORDER_NO");
&I = 1;

/* Create record objects for the Header and Lines */
&ln_rec = CreateRecord(Record.DEMAND_INF_INV);
&hdr_rec = CreateRecord(Record.MSR_HDR_INV);

/* Loop through each Header row that is fetched */
While &hdr_sql.Fetch(&hdr_rec)
   /* Publish the message if it's size exceeds the MaxMessageSize specified in
Utilities/Use/PeopleTools Options */
   If &STOCK_MSG.Size > %MaxMessageSize Then
      &STOCK_MSG.Publish();
      &I = 1;
      /* Create a new instance of the message object */
      &STOCK_MSG = CreateMessage(Message.STOCK_REQUEST);
      &hdr_rs = &STOCK_MSG.GetRowset();
   End-If;
   If &I > 1 Then
      &hdr_rs.InsertRow(&I - 1);
   End-If;
   /* Instantiate the row within the Header portion of the
      App Message rowset to which data will be copied */
   &hdr_rec_msg = &hdr_rs.GetRow(&I).GetRecord(Record.MSR_HDR_INV);
   /* Copy data into the level 0 (Header portion) of &STOCK_MSG message
structure */
   &hdr_rec.CopyFieldsTo(&hdr_rec_msg);

   /* Publish the last message if it has been changed*/
   If &hdr_rec_msg.IsChanged Then
      &STOCK_MSG.Publish();
   End-If;
End-While;
```

XmlDoc Class Inbound Asynchronous Example

The following example uses the GetMessageXmlDoc built-in function.

```
Local XmlDoc &BIGMAN;
Local XmlNode &node, &root;
```

```
Local string &outstring;
Local Rowset &LEVEL0;
Local Record &SALES_ORDER_INFO, &REC;

&CRLF = Char(13) | Char(10);

&BIGMAN = GetMessageXmlDoc();

&root = &BIGMAN.DocumentElement;
&child_count = &root.ChildNodeCount;

/* Get values out of XmlDoc */
&node_array = &root.GetElementsByTagName("QE_ACCT_ID");
&acct_id_node = &node_array.Get(2);
&account_id_value = &acct_id_node.NodeValue;

&node_array = &root.GetElementsByTagName("QE_ACCOUNT_NAME");
&acct_name_node = &node_array.Get(2);
&account_name_value = &acct_name_node.NodeValue;

&node_array = &root.GetElementsByTagName("QE_ADDRESS");
&address_node = &node_array.Get(2);
&address_value = &address_node.NodeValue;

&node_array = &root.GetElementsByTagName("QE_PHONE");
&phone_node = &node_array.Get(2);
&phone_value = &phone_node.NodeValue;

&outstring = "GetMessageXmlDoc Test";
&outstring = &outstring | &CRLF | &account_id_value | &CRLF |
&account_name_value | &CRLF | &address_value | &CRLF | &phone_value;

&SALES_ORDER_INFO = CreateRecord(Record.QE_SALES_ORDER);
&SALES_ORDER_INFO.GetField(Field.QE_ACCT_ID).Value = &account_id_value;
&SALES_ORDER_INFO.GetField(Field.DESCRLONG).Value = &outstring;
&SALES_ORDER_INFO.Update();
```

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Understanding PeopleCode and Events"

PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

Handling Inbound Synchronous Transactions

Use the GetMessage or GetMessageXmlDoc built-in function, and place your PeopleCode in the message OnRequest event.

When you receive an inbound request message, your response message must conform to the return type expected by the sending code. If the sender uses the Message class SyncRequest method, you must respond with a Message object. If the sender uses the SyncRequestXmlDoc function, you must respond with an XmlDoc or SoapDoc object.

Message Class Inbound Synchronous Example

The following example uses the GetMessage built-in function.

```
Local Message &request, &response;

&request = GetMessage();
&response = CreateMessage(Message.QE_SO_SYNC_RESP);
&response.GetRowset().GetRow(1).GetRecord(Record.QE_SO_RESPONSE).GetField(Fiel
d.DESCRLONG).Value = "He's a nice person ";

ReturnToServer(&response);
```

XmlDoc Class Inbound Synchronous Example

The following example uses the GetMessageXmlDoc built-in function.

```
Local XmlDoc &xmlRequestDoc;
Local XmlDoc &xmlResponseDoc;
Local XmlNode &select;
Local array of XmlNode &whereClause;

Local string &recordName;
Local string &fieldName;
Local string &fieldValue;
Local Rowset &outputRowset;
Local boolean &b;

&xmlRequestDoc = GetMessageXmlDoc();
&select = &xmlRequestDoc.DocumentElement;

&recordName = &select.GetAttributeValue("record");
&outputRowset = CreateRowset(@("Record." | &recordName));

&whereClause = &select.GetElementsByTagName("where");
If &whereClause <> Null And
      &whereClause.Len <> 0 Then
   &fieldName = &whereClause.Get(1).GetAttributeValue("field");
```

```
   &fieldValue = &whereClause.Get(1).GetAttributeValue("value");
   &outputRowset.Fill("WHERE " | &fieldName | "= :1", &fieldValue);
Else
   &outputRowset.Fill();
End-If;

&xmlResponseDoc = CreateXmlDoc();
&b = &xmlResponseDoc.CopyRowset(&outputRowset);
ReturnToServer(&xmlResponseDoc);
```

SoapDoc Class Inbound Synchronous Example

The following example uses the GetMessageXmlDoc built-in function.

Note. Because GetMessageXmlDoc is an XmlDoc function, you must receive the inbound request message as an XmlDoc object, then convert it to a SoapDoc object to process it using SOAP methods. Likewise, because the request was sent using SyncRequestXmlDoc, you must convert your SoapDoc response message to an XmlDoc object before returning it.

```
Local XmlDoc &request, &response;
Local string &strXml;
Local SoapDocs &soapReq, &soapRes;

&soapReq = CreateSoapDoc();

&request = GetMessageXmlDoc();
&soapReq.XmlDoc = &request;
&OK = &soapReq.ValidateSoapDoc();
&parmN = &soapReq.GetParmName(1);
&parmV = &soapReq.GetParmValue(1);

&soapRes = CreateSoapDoc();
&soapRes.AddEnvelope(0);
&soapRes.AddBody();
&soapRes.AddMethod("StockPrice", 1);
&soapRes.AddParm(&parmN, &parmV);
&soapRes.AddParm("Price", "29");
&OK = &soapRes.ValidateSoapDoc();

&response = &soapRes.XmlDoc;
ReturnToServer(&response);
```

See Also

PeopleTools PeopleBook: PeopleCode Developer's Guide, "Understanding PeopleCode and Events"

PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

PeopleTools PeopleBook: PeopleCode Reference, "SOAPDoc Class"

Processing Inbound Errors

Several kinds of errors can occur during the sending or receiving process. You can monitor these errors, and in many cases fix them in Integration Broker Monitor.

This section explains how to:

- Validate data.
- Use the PeopleCode Exit function.
- Correct messaging errors.

See Also

Using the Exit Function

Validating Data

XMLDoc and SoapDoc Class Validation

- You can validate incoming XML DOM compliant messages using the XmlDoc document type definition (DTD) delivered with your PeopleSoft application. See *PeopleTools PeopleBook: PeopleCode Reference,* "XmlDoc Class."
- If your inbound message is SOAP compliant, you can validate it using the PeopleCode SoapDoc class method ValidateSoapDoc. See *PeopleTools PeopleBook: PeopleCode Reference,* "SOAPDoc Class."

Message Class Validation

A message receiving process can validate incoming data in the following ways:

- Use the **ExecuteEdits** method on the message to invoke the definitional edits.
- In cases where PeopleCode validation functions already exist, for example in a FUNCLIB, or validation logic can be encapsulated within a small set of functions. You can call these functions from within the receiving PeopleCode.
- For complex validation logic, you can call a component interface or Application Engine program from your receiving process. This allows you to re-use logic embedded in the component.

The **ExecuteEdits** method uses the definitional edits to validate the message. The following system variables can be specified alone or in combination. Not specifying any variable with the **ExecuteEdits** method executes all the edits.

- %Edit_DateRange
- %Edit_OneZero
- %Edit_PromptTable
- %Edit_Required
- %Edit_TranslateTable
- %Edit_YesNo

The following example executes all the edits for all levels of data in the message structure:

```
&MYMSG.ExecuteEdits();
```

The following example executes the Required Field and Prompt Table edits:

```
&RECPURCHASEORDER.ExecuteEdits(%Edit_Required + %Edit_PromptTable);
```

ExecuteEdits uses *set processing* to do the validation. Validation using a component interface or a PeopleCode function is usually done with row by row processing. If a message consists of a large number of rows per rowset, consider writing the message to a staging table and calling an Application Engine program to do set processing if you want additional error checking.

The ExecuteEdits methods sets several properties on several objects if there are any errors.

- **IsEditError** is set on the Message, Rowset, Row, and Record object if there are any fields that contain errors.
- **EditError, MessageNumber** and **MessageSetNumber** are set on the field object that contains the error.

If you don't want to use ExecuteEdits, you can set your own errors using the field properties. Setting the **EditError** property to True automatically sets the message property **IsEditError** to True. You can also specify your own message number, message set number, and so on, for the field. When you finish setting the fields that are in error and if you used the **Exit(1)** built-in function, the message status changes to **Error**.

Using the Exit Function

Use the **Exit** function to invoke a messaging error process when the application finds an error. This works only when you use the PeopleCode Message class to process inbound transactions. The same error processing is invoked automatically if PeopleTools finds an unexpected error, for example an SQL error. The Exit function has an optional parameter which affects how the error is handled.

If you want to handle error processing in your application tables, use the Exit function with no parameter, or just let the subscription process complete normally. This marks the message receipt as successful and commits.

If you want Integration Broker to handle the error tracking and correction, you must use the Exit function with 1 as a parameter, to log the errors, perform a rollback, and stop processing.

Exit( )

In the Exit( ) form, everything is committed and the message is marked as complete. You can use this to indicate that everything processed correctly. However, sometimes you may want to use this function to stop program processing. Note, though, that the message status is set to complete; therefore you can't detect or access errors from the Integration Broker Monitor. If errors did occur, the subscription code should write them to a staging table, and then you would have to handle all of the error processing.

The Exit function does the following:

- Sets the message status in the application message queue for the Subscription to Done.
- Commits the transaction.
- Stops execution.

Sample Exit( ) PeopleCode:

```
&Msg.ExecuteEdits();
If &Msg.IsEditError then
   App_Specific_Error_Processing();
   Exit();
Else
   Specific_Message_Processing();
End-if;
```

Exit(1)

In this form, the Exit function does the following:

- Executes a rollback.
- Sets the message status in the message queue for the subscription to **error.**
- Writes the errors to the subscription contract error table.
- Stops execution.

You can view all errors that have occurred for this message in the Integration Broker Monitor, even those errors detected by ExecuteEdits.

Sample Exit(1) PeopleCode:

```
&Msg.ExecuteEdits();
If &Msg.IsEditError then
```

```
   Exit(1);
Else
   Process_Message();
End-if;
```

See Also

PeopleCode Reference Guide, "PeopleCode Built-In Functions," Exit

"Using Integration Broker Monitor"

Correcting Errors

If you anticipate having only a few errors to correct, you can do so in the Message Errors page of the Integration Broker Monitor.

If you have many errors, you may want to create a page that a user could use to correct the data for a particular message. Generally, your user will be correcting errors that were detected when running the Message class method ExecuteEdits or another function as part of your receiving code.

For the search record for your page, you may want to create a view based on the MSGNAME and SUBCONSTATUS fields of the PSAPMSGSUBCON. MSGNAME lists all the messages that have subscription contracts. The SUBCONSTATUS field has the following values:

Status	*Value*
ERROR	0
NEW	1
STARTED	2
WORKING	3
DONE	4
RETRY	5
TIMEOUT	6
EDITED	7
CANCELLED	8

If you create a page, you must specify it in your message definition properties. After you define it, you can either use it as a regular page, or access it from the Message Details link in the Integration Broker Monitor.

Correcting Message Errors in PeopleCode

Example 1

The following code is located in the PreBuild event of a page that's used to fix message errors. It gets the inbound message as specified by the search record (PSAPMSGSUBCON) and loads the fields of a work record (MS_EDIT_WRK) with the data from the message:

```
Local Rowset &RS_PLAYER;
Local Message &MSG_EDIT_TEST;

&MSG_EDIT_TEST = GetSubMessage(PSAPMSGSUBCON.PUBID, PSAPMSGSUBCON.PUBNODE,
PSAPMSGSUBCON.CHNLNAME, PSAPMSGSUBCON.MSGNAME, PSAPMSGSUBCON.SUBNAME);

&RS_PLAYER = &MSG_EDIT_TEST.GetRowset();

MSG_EDIT_WRK.QE_PLAYER_ID = &RS_PLAYER(1).QE_PLAYER.QE_PLAYER_ID.Value;

MSG_EDIT_WRK.QE_PLAYER_STATUS =
&RS_PLAYER(1).QE_PLAYER.QE_PLAYER_STATUS.Value;

MSG_EDIT_WRK.DESCRLONG = &RS_PLAYER(1).QE_PLAYER.DESCRLONG.Value;
```

Example 2

The following code is located in the SavePostChange event. It takes the values from the work record and repopulates the rowset. As the data that is in the rowset is identical to the data in the message, this changes the values of the fields in the message. When you use the Update method, the status of the message in the Integration Broker Monitor is automatically changed to Edited.

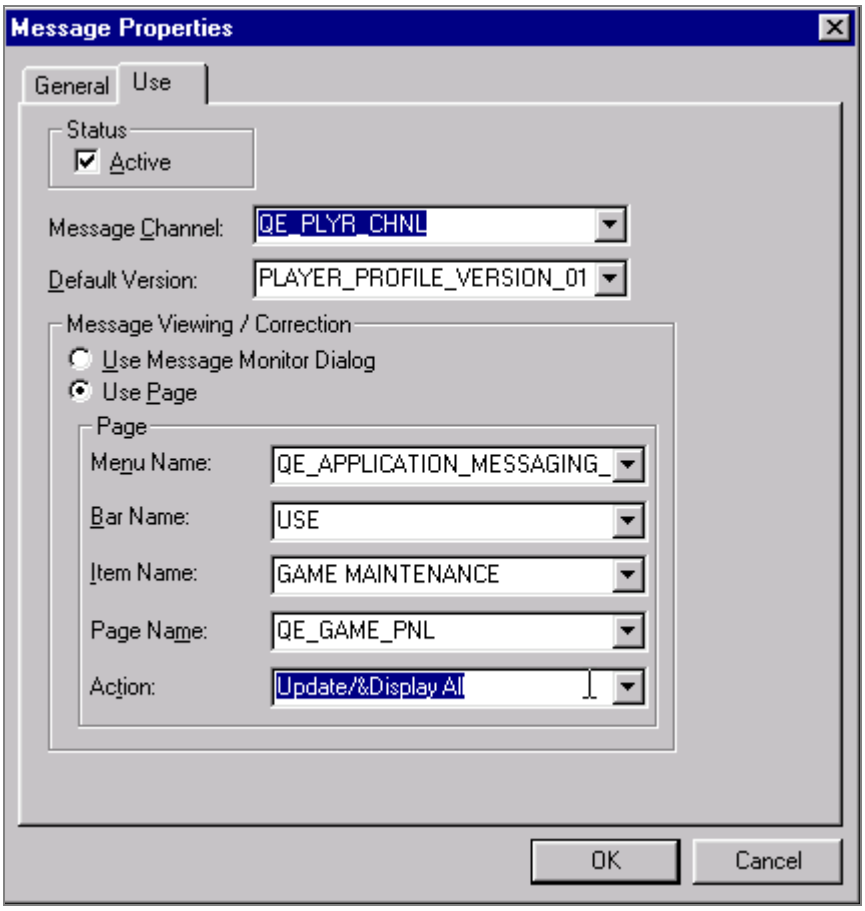
```
Local Rowset &RS_PLAYER;
Local Message &MSG_EDIT_TEST;

&MSG_EDIT_TEST = GetSubMessage(PSAPMSGSUBCON.PUBID, PSAPMSGSUBCON.PUBNODE,
PSAPMSGSUBCON.CHNLNAME, PSAPMSGSUBCON.MSGNAME, PSAPMSGSUBCON.SUBNAME);

&RS_PLAYER = &MSG_EDIT_TEST.GetRowset();

&RS_PLAYER(1).QE_PLAYER.QE_PLAYER_ID.Value = MSG_EDIT_WRK.QE_PLAYER_ID;

&RS_PLAYER(1).QE_PLAYER.QE_PLAYER_STATUS.Value =
MSG_EDIT_WRK.QE_PLAYER_STATUS;

&RS_PLAYER(1).QE_PLAYER.DESCRLONG.Value = MSG_EDIT_WRK.DESCRLONG;

&MSG_EDIT_TEST.Update();
```

Correcting Data Errors in a Page

If you want to use a pre-defined page to correct data errors, you specify that in message properties.

To correct data subscription contract errors in a page:

1. In Application Designer, open your message definition or define a new one.
2. Go to the **Message Properties** screen, **Use** tab.
3. Click **Use Page**, and specify page characteristics.



Specify page for error correction

4. Enter the menu name, bar name, item name, page name, and action from the dropdown lists.

To launch the page

1. Open the Message Monitor.

   Select the Sub Contracts page and highlight the message in error.

2. Click Message Details to view the message details.

See Also

"Using Integration Broker Monitor"

CHAPTER 7

Administering Basic Integrations

This chapter covers basic management of gateways, target connectors, nodes and transactions, and discusses how to:

- Configure a gateway.
- Configure a node.
- Configure transactions.

Understanding Integration Administration

The modular architecture of Integration Broker separates the integration development activity from the administrative activity. You should already have defined your message channel, messages, and sending and receiving PeopleCode in Application Designer. Having developed those basic elements, you'll use the Integration Broker PIA components to employ them in your current integration. See "Creating and Implementing Integrations," Producing Basic Integrations for an overview of this process.

Administrators use these components to specify how an outbound or inbound message should be treated, including where the message should be sent, how the message should be modified, and what criteria should be used for these actions. An administrator essentially defines an integration by specifying information about the gateway, connectors, nodes, transactions and relationships to be used in the integration.

When the Integration Engine receives a message, it searches the transaction definitions associated with that message definition to determine the valid target nodes for the message, then passes the message to the synchronous or asynchronous request handler as appropriate. The handler directs the message to the specified nodes, in each case using the gateway and target connector specified by the node definition. It could also direct the message to local Subscription or OnRequest PeopleCode. A transaction might also be modified by a defined relationship which could involve a transformation. See "Creating and Implementing Integrations," Producing Advanced Integrations for an overview of relationships and transformations.

Note. To configure a node, at least one gateway with one connector must be defined. When you define a gateway, node or transaction, *current* refers to the definition you have open — for example, the *current node. Selected* refers to an item you've indicated other than the definition you have open — for example, the *selected transaction.*

See Also

"Understanding Integrations"

"Creating and Implementing Integrations"

Configuring a Gateway

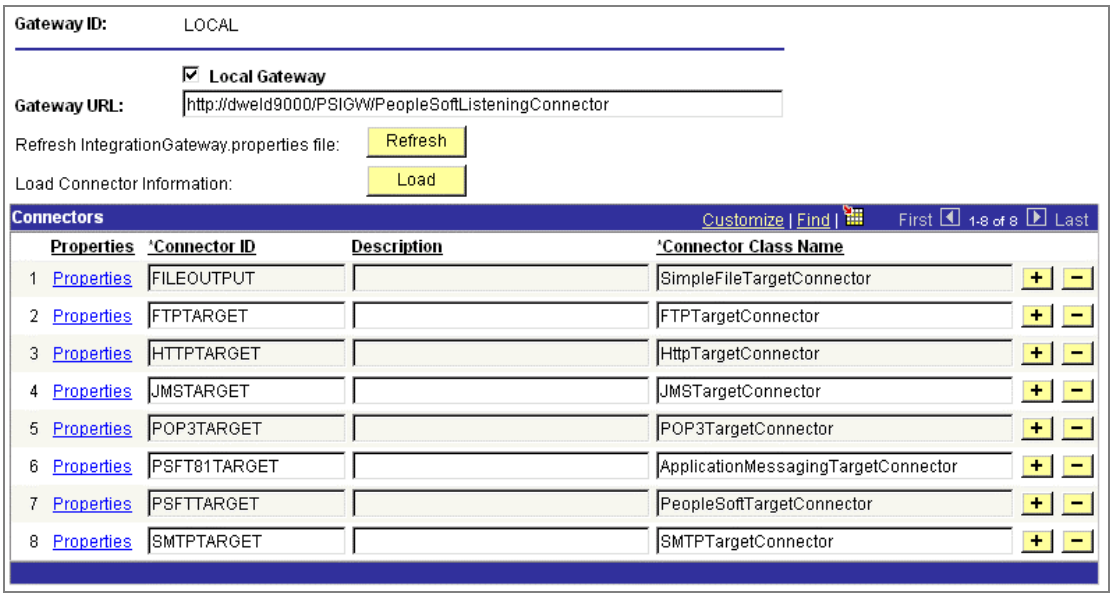
This section explains how to:

- Define a new gateway.
- Specify the gateway location.
- Refresh the gateway properties.
- Register installed target connectors associated with the gateway.
- Specify available properties for a connector.

Understanding Gateway Configuration

You configure a gateway using the Gateways component, which consists of a single page. Use this page to specify the gateway's location, update configuration settings, and register target connectors to be used with the gateway. To configure a node, at least one gateway with one connector must be defined.

Note. A default definition for your local gateway is automatically created upon installation. If you plan to use only the local gateway, there's no need to create a new definition, but you still must configure the gateway.



Gateways page

Note. You typically register and configure the gateway target connectors only when you install a new gateway or a new connector.

See Also

"Understanding Integrations"

Defining a New Gateway

To define a new gateway:

1. Select PeopleTools, Integration Broker, Gateways.

   The Gateways search page appear.

2. Add a new value, enter the integration gateway ID for your new gateway and click the **Add** button.

   The Gateways page appears.

   Note. The Integration Gateway ID for the delivered local gateway is ***LOCAL.***

Specifying the Gateway Location

To specify the gateway location:

1. Open the desired gateway definition.

   Select **PeopleTools, Integration Broker, Gateways,** and enter the Integration Gateway ID. The Gateways page appears.

2. (Optional) Select Local Gateway to designate the gateway as local.

   Each Integration Broker node requires exactly one local gateway, which is your application's first point of contact with other PeopleSoft applications, third parties, Integration Broker hubs, and remote gateways.

   Note. To change which gateway is the local one, you must open the definition of the currently designated local gateway and clear its **Local Gateway** check box before you can select it in another definition.

3. Enter the gateway URL for the selected gateway's PeopleSoft listening connector:

   `http://<gatewayservername>/PSIGW/PeopleSoftListeningConnector`

   The gateway uses the PeopleSoft listening connector to receive messages from an Integration Engine node or a remote gateway.

Refreshing the Gateway Properties

The behavior of an installed integration gateway is governed by settings in the configuration file **IntegrationGateway.properties**. When the gateway server boots, it reads and applies the properties in this file. However, changes you make to the file while the server is running have no immediate effect.

Integration Broker provides a solution for this difficulty. Click the **Refresh** button on the Gateways page, and Integration Broker reapplies the configuration settings in the current gateway's IntegrationGateway.properties file to the gateway without rebooting, including any changes you've made.

Note. For the JMS properties, the Refresh button has no effect. You must stop and restart the JMSListeningConnectorAdministrator servlet to refresh its properties.

See Also

"Using the Integration Gateway," Setting Integration Gateway Properties

"Using the Integration Gateway," Setting JMS Configuration Properties

Registering Installed Target Connectors

The Connectors grid on the Gateways page lists the connectors registered with the current gateway. Integration Broker provides two ways of registering target connectors with the gateway — automatically by introspection, or manually by entering information directly.

Registering Connectors by Introspection

If the connector was delivered with your PeopleSoft application, or developed using PeopleSoft's Integration Broker Connector Software Development Kit (SDK), you can easily register it with Integration Broker's connector introspection feature. Before you can register a new connector, you must install it — see "Using the Integration Broker Connector SDK."

Click the **Load** button on the Gateways page to trigger the introspection for the current gateway, whether it's local or remote. Integration Broker examines all target connectors installed on the gateway to determine their properties, and loads those properties into the gateway definition. All of the connectors appear in the Connectors grid, and the properties of each connector are updated to reflect the connector's current state.

Note. The introspection never overrides existing information. Rather, it supplements missing information from your Integration Gateway configuration, so any manually edited values won't be affected. Also, if you modified a connector, any new and modified properties are loaded without interfering with the properties that currently exist.

Registering Connectors Manually

To register and configure a connector manually, you must enter the Connector ID, Connector Class Name, and property information that's hardcoded in the connector. This information is provided by PeopleSoft for all delivered connectors; information about connectors from any other source must be provided by that source.

To register a new connector:

1. Click the **Add a new row** button in one of the rows on the Connectors grid of the Gateways page.

   A new empty row appears below the row you clicked.

   Note. If the gateway has no registered connectors, an empty row is already available.

2. Enter the connector ID for your new connector.
3. Enter the connector class name.
4. Click **Properties** to edit the connector's properties.

   The Properties page for the selected connector appears. See Editing Connector Properties.

Note. You can remove a connector entry from the Gateways page by clicking the **Delete row** button in its row.

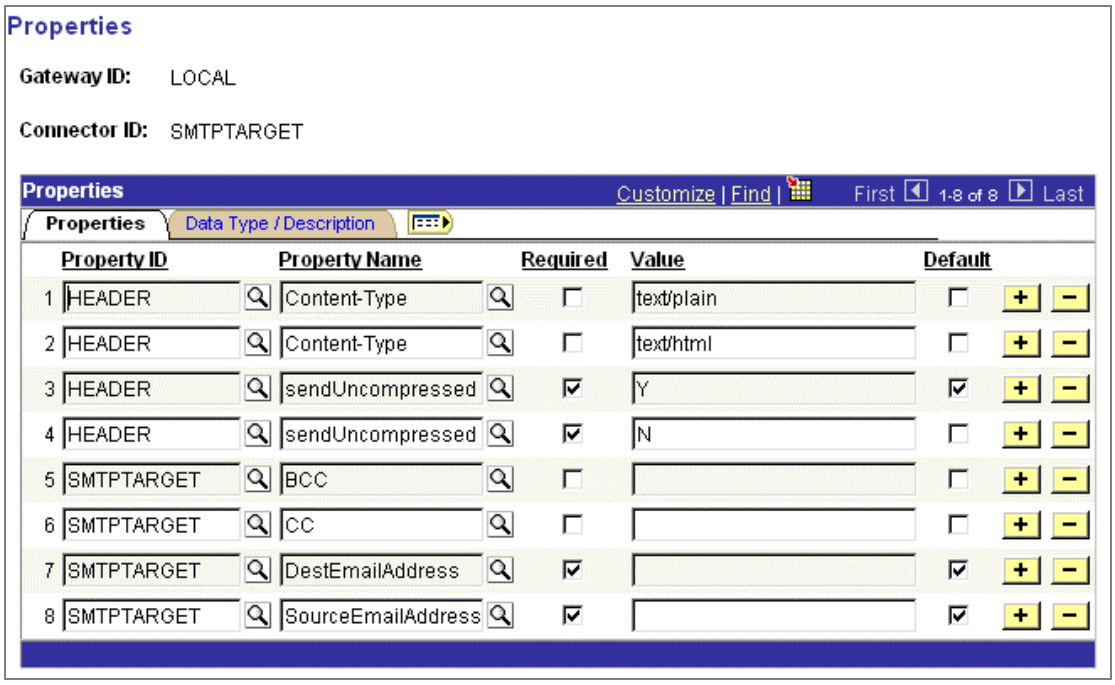
See Also

"Using the Integration Gateway," Working with Target Connectors

Editing Connector Properties

Every target connector has a set of properties that represent various parameters which may be used by the connector. These properties are hardcoded in the connector class. The Connector Properties page is a master list of all of a connector's available properties and their configurations. When you specify the connector to use in a node definition, a subset of the properties on this master list is invoked by the node.

Note. All available properties of a connector are automatically entered on the Connector Properties page when you register the connector.



Gateways — Properties page: Properties tab

Each property entry is defined by a combination of property ID and property name, both of which must already exist in the connector class. A single connector can be used to process messages that adhere to a variety of different header formats, communication protocols or other requirements. You can represent all of these variations on the Connector Properties page by entering multiple instances of the properties used, each with a different value.

You may find it necessary to add a property entry if your integrations that use a given connector frequently require one of its properties to have a different value from the ones shown on the Connector Properties page. You can add a new instance of the property and provide the new value, which is then available on the value lookup table when you configure a node.

To add a new property instance:

1. Click the **Add a new row** button in one of the rows of the Connector Properties page.

   A new empty row appears below the row you clicked.

   Note. If the connector has no properties entered, an empty row is already available.

2. Select a property ID.

   The available Property IDs are specific to the connector you're configuring.

3. Select a property name.

   The available Property Names are specific to the Property ID you selected.

4. If this property is required for the connector to work properly, select the Required check box.

   You can clear the Required checkbox, but this isn't recommended if the connector actually requires this property. It makes sense for all instances of a property (a Property ID/Property Name combination) to have the same Required status.

5. Enter an appropriate value for the property instance.

   Appropriate Values might come from PeopleSoft, from the connector's developer, or from your own experience and requirements.

6. Select the Default check box for one or more of the required property instances.

   When you specify this connector in a node definition, only the connector properties marked as *both* required and default appear automatically on the Connectors page of the Node Definitions component. See Specifying a Connector.

   Note. In most cases, only one instance (value) of a required property should be used by a given node, but you may want to designate multiple values as default so they all appear. You must keep in mind which properties can be used with multiple values, and which require a single value.

7. Save the properties.

8. Select Return to Connectors.

   The Gateways page appears.

See Also

"Using the Integration Broker Connector SDK"

Configuring a Node

This section explains how to:

- Define a new node.
- Specify general node information.
- Specify contact information.
- Define node properties.
- Specify a gateway and connector.
- Add connector properties.
- Modify connector property values.

Understanding Node Configuration

You configure a node using the Node Definitions component, which consists of several pages.

Local and Remote Nodes

When discussing nodes, it's important to consider the concepts of local and remote nodes. Whether a node is local or remote is determined by which database the node is defined in. If you're signed on to database A which has node A defined, then node A is local. Later, if you're signed on to database B, node A is remote. Therefore local and remote are relative terms.

Note. In practice, only portals use nodes designated *Local*. The only local node used by Integration Broker is designated *Default Local*, which means the same thing as *Local*, but applies only to Integration Broker. See Specifying General Node Information.

Defining a New Node

To create a new node definition:

1. Select PeopleTools, Integration Broker, Node Definitions.

   The Nodes search page appears.

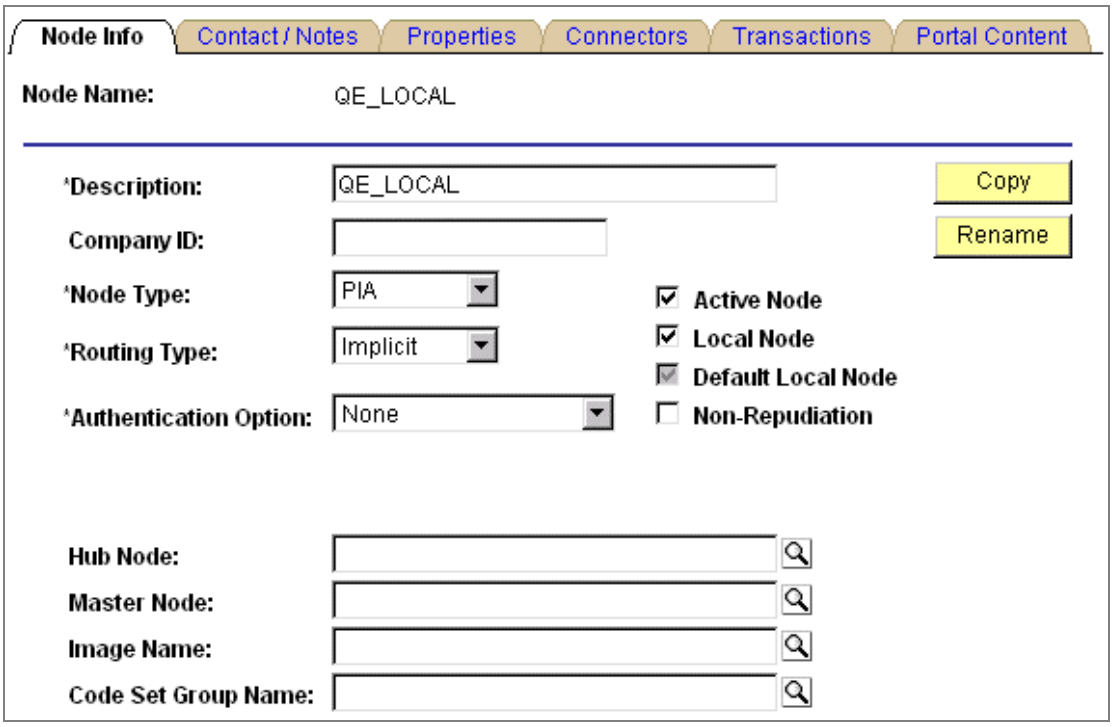
2. Add a new value, enter a node name for your new node and click **Add.**

   The Node Info page appears.

Note. The name you specify for a remote node must be the same as the name it specifies for itself. If you want to configure an existing node, enter the node name on the search page.

Specifying General Node Information

Access the Node Definitions – Node Info page.



Node Definitions – Node Info page

Description Enter a descriptive name for the node.

Company ID (Optional) The name of the company or organization associated with this node.

Node Type Select from the following types:

PIA: This designates the node as a PeopleSoft database using Integration Broker. This is the default for a new node.

External: This designates the node as an entity that doesn't use Integration Broker.

ICType: This is a portal-specific setting that Integration Broker doesn't use.

Routing Type

The routing type specifies how the Integration Engine should make the determination of whether to send a message to this node. Select from the following types:

Explicit: The current node won't be included as a target node unless you specify it directly in your sending PeopleCode. You can do this in the following ways:

- Specify the target node as a parameter of the message class Publish or SyncRequest methods. See *PeopleTools PeopleBook: PeopleCode Reference,* "Message Class."
- Specify the target node as a parameter of the PublishXMLDoc or SyncRequestXMLDoc built-in functions. See *PeopleTools PeopleBook: PeopleCode Reference,* "XmlDoc Class."

If you selected a node type of *External,* the routing type is *Explicit* by default.

Implicit: All nodes with this routing type are included as target nodes unless your sending PeopleCode references specific ones. In that case, only the referenced nodes are targets. If you selected a node type of *PIA,* the routing type is *Implicit* by default.

Note. The routing type can be overridden for individual outbound transactions. See Editing Transaction Details.

Authentication Option

Select from the following options:

Certificate: This node inserts a digital certificate in messages it sends, and expects messages it receives to include a digital certificate.

With a PIA node, upon receiving the message, Integration Broker extracts the distinguished name from the certificate and validates it against the distinguished name retrieved from this node's keystore file. Messages sent by this node have the digital certificate automatically inserted by Integration Broker. See *PeopleTools PeopleBook: Security,* "Setting up Digital Certificates and Single Signon," Working With Digital Certificates.

If this node is external, it's expected to handle certificates the same way as a PIA node.

None: No authentication is required by this node. This is the default value.

Password: Two new fields appear: **Password** and **Confirm Password**. Enter your password in the first edit box, and confirm it in the second edit box.

With a PIA node, Integration Broker expects messages both outbound to and inbound from this node to include a password, which it validates against the password entered here.

If this node is external, it's expected to handle passwords the same way as a PIA node.

Active Node Select this check box to make this node active, so it can be used by Integration Broker.

Note. If you clear this check box to deactivate the node, any transactions and relationships that use it are also deactivated. Activating the node won't automatically reactivate the transactions and relationships; they must be individually reactivated. See Editing Transaction Details and "Administering Relationships."

Local Node Clear this check box for all Integration Broker nodes except the default local node. Any nodes defined as local without being the default local node are applicable only to portals, and aren't used by Integration Broker. Integration Broker uses remote nodes (where Local Node = *0* on the search list) or the default local node (where Default Local Node = *Y*).

Default Local Node This check box is read-only, and indicates that the current node represents the database you're signed on to. Integration Broker is delivered with one node predefined as the default local node.

Note. You can rename the default local node to more appropriately reflect your application or system, using the **Rename** button.

Non-Repudiation Select this check box to activate nonrepudiation functionality for this node.

Note. You must also activate nonrepudiation in the definition of each message for which you want this feature. See "Defining Messages," Understanding Nonrepudiation.

Rename Click Rename to change the name of the current node. The Rename Node page appears, prompting you to enter a new name for the node.

Copy Click Copy to define a new node with exactly the same properties as the current node. The Save Node As page appears, prompting you to enter a name for the new node. The Default Local check box is cleared for all new nodes.

	Note. If you copy a local node, the new node will be local as well. You must clear the Local Node check box to use it with Integration Broker.
Delete	Click Delete to delete this node definition. The Delete Node page appears. Click Delete again, and you're prompted to confirm the deletion. **Note.** You cannot delete the default local node.
Hub Node	(Optional) Select the name of a node that will serve as a "gatekeeper" node for the current one. All transactions outbound from the default local node to the current node will be redirected to the selected hub node instead. This way you can add hub capability for existing transactions that specify the current node as a target, without having to modify those transactions. See "Administering Relationships" for more information about hub nodes.
Master Node	(Optional) This field is informational, and complementary to the hub node designation. If the current node is used as a hub, you can indicate here the target node with which it's associated. If the current node represents a subordinate database, you can indicate the primary database.
Image Name	(Optional) Select an image from the system database. Any application that uses images can use the selected image to represent the current node.
Code Set Group Name	(Optional) Select the codeset group to which you want the current node to belong. Transform programs invoked by the default local node use this association to search for message data requiring translation. See "Applying Transformation, Translation and Filtering."

Corresponding Node Definitions

Each Integration Broker node involved in an integration must contain a default local node definition for itself, and a remote node definition for each of the other nodes involved.

Thus, if the following definitions exist in the node **A** database:

- NODE_A (default local)
- NODE_B (remote)

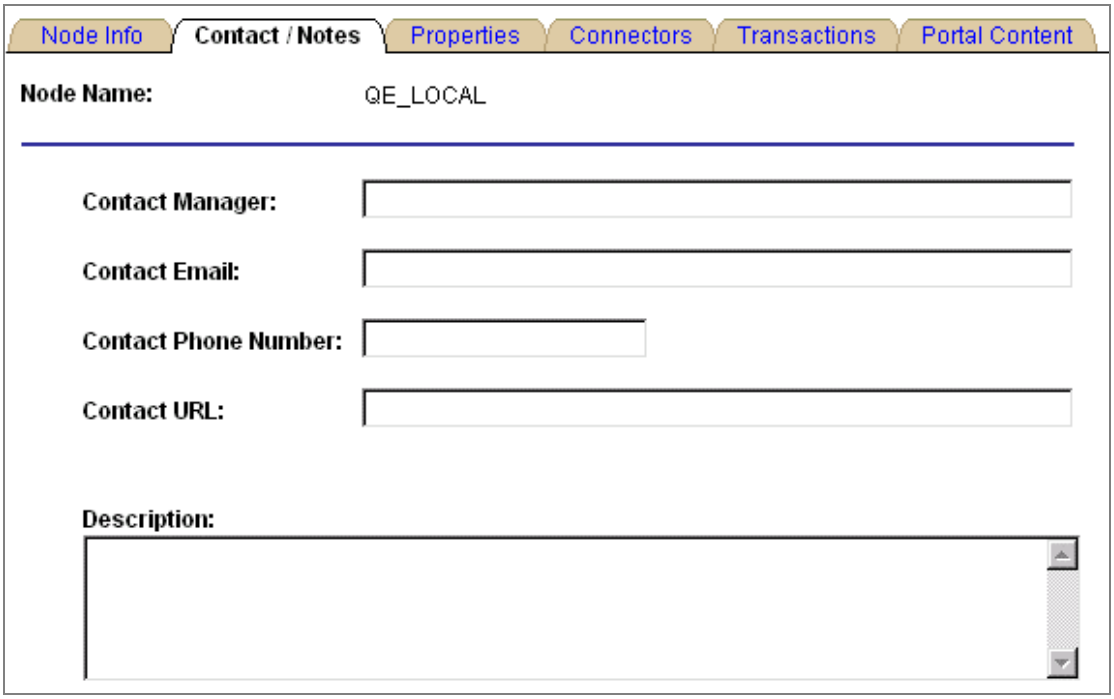
The following definitions must exist in the node **B** database for it to integrate with node A:

- NODE_A (remote)
- NODE_B (default local)

Specifying Contact Information

Access the Node Definitions – Contact/Notes page.

Each node represents a database or other software entity managed by one or more persons. You can use this page to record useful information about the person or people associated with this node.



Node Definitions – Contact/Notes page

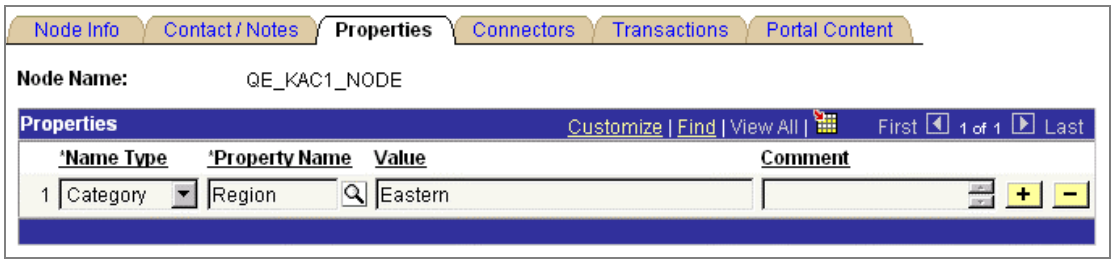
Contact Manager	The name of the representative or administrator of the node, in standard PeopleSoft name format.
Contact Email	The Contact Manager's email address, in standard PeopleSoft email address format.
Contact Phone Number	The Contact Manager's telephone number.
Contact URL	The address of the Contact Manager's support web site, if there is one.

Defining Node Properties

Access the Node Definitions – Properties page.

This page provides a convenient place to store additional information about a node that can be referenced by any other node. Properties created for all nodes are stored in a single table, PSNODEPROP.

Examples include a DUNS number or Tax Identification Number. These properties can be used to update messages with additional information. They can also serve to add additional categorization for custom processing; for example, add a *Region* property so nodes can be referenced by region for some special processing.



Node Definitions — Properties page

Name Type Select from the following types:

Category: Indicates that the property is used for categorization.

Ident: Indicates that the property is used for identification.

Search: Indicates that the property is used for searching.

Property Name Enter a new property name or select an existing property of the selected name type.

Value Enter the desired value for the selected property.

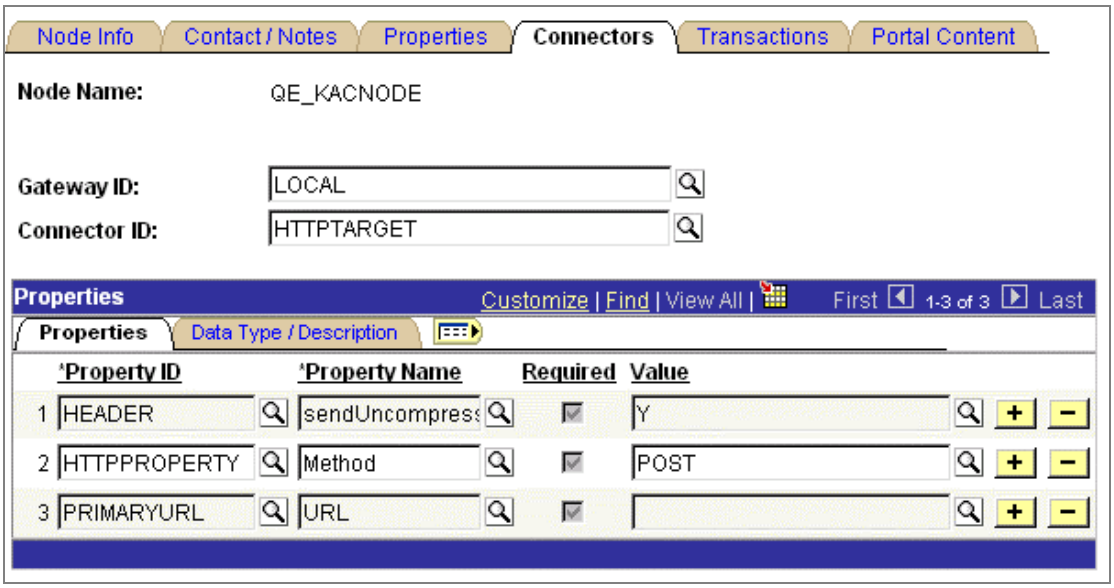
Specifying a Connector

Access the Node Definitions – Connectors page.

Use this page to specify how the local default node should send messages to the current node, including which gateway, which target connector registered with that gateway, and which property settings for that connector should be used.

You specify the current node's local gateway and target connector on this page. If the current node is remote, it can use the default local node's gateway or any other installed gateway as its local gateway.

At least one gateway with at least one target connector must already be defined and configured. If the current node has its own gateway installed, the default local node's database must contain a definition for it, configured as a remote gateway. See Configuring a Gateway.



Node Definitions — Connectors page: Properties tab

To specify a connector for this node:

1. Select the gateway ID for the gateway you want this node to use.

   When the default local node sends a message to any other node, the message first goes to the default local node's local gateway through its PeopleSoft listening connector, regardless of the gateway ID you select here.

   - If you specify a remote gateway ID, the local gateway uses its default remote gateway connector (specified in the IntegrationGateway.properties file) to route messages to the remote gateway through the remote gateway's PeopleSoft listening connector. The remote gateway sends the messages directly to the current node, using the connector you specify in the next step.

     Note. The default remote gateway connector setting initially specifies the HTTP target connector, which is unlikely to change unless you develop a custom target connector for this purpose. See "Using the Integration Gateway," Setting Integration Gateway Properties.

   - If you specify the local gateway ID, the local gateway sends messages directly to the current node, using the connector you specify in the next step.

2. Select a connector ID from the list of connectors registered with the selected gateway.

   This field specifies the target connector appropriate to the communication method preferred by the current node. If the node is a PeopleSoft application with Integration Broker installed, select *PSFTTARGET*. If the node is a PeopleSoft 8.1x application, select *PSFT81TARGET*.

The rows on the Properties and Data Type / Description tabs are automatically populated with the connector's properties that are designated Required in the gateway definition. The fields on these tabs are the same as those on the Connector Properties page. If the connector has multiple instances of a required property defined, only the instance designated as Default appears. See Editing Connector Properties.

Note. You can override the gateway and connector selection for individual outbound transactions; see Editing Transaction Details.

Modifying Connector Properties for this Node

Any properties that appear on this page are only a starting point. They're copies of the specified connector's original properties, and you can modify them here without fear of destroying those originals. You can add another property, modify a property's value and description, or remove a property. Information about appropriate modifications might come from PeopleSoft, from the connector's developer, or from your own experience and requirements.

Important! Take care not to remove a required property unless you replace it with another instance of the same property. Without all of its required properties, the connector is unlikely to work correctly.

Adding Connector Properties

To add another property:

1. Click the **Add a new row** button in one of the rows on the Properties grid of the Connectors page.

   A new empty row appears below the row you clicked.

   Note. If the connector has no properties entered, an empty row will already be available.

2. Select a property ID.

   The available Property IDs are specific to the connector you're configuring.

3. Select a property name.

   The available Property Names are specific to the Property ID you selected.

Modifying Property Values

To modify a property's value and description:

1. Ensure that the property has the correct value for your planned use.

   You can select an existing Value from the available property instances, or enter a new Value.

2. (Optional) Enter a description of the property instance.

   It's a good idea to modify the property's description to reflect any change in its Value.

Note. You can remove a Property entry by clicking the **Delete row** button in its row.

See Also

Editing Connector Properties

"Using the Integration Gateway," Working with Target Connectors

Configuring Transactions

This section explains how to:

- View the transaction list.
- Define a transaction.
- Edit transaction details.
- Edit transaction message details.
- Edit transaction connector properties.

Understanding Transactions

A transaction is the basic unit of work in an integration. Each transaction is associated with a specific node. You use a transaction to designate a message that the current node can send or receive, and whether that message will be inbound or outbound, and whether it will be synchronously or asynchronously transmitted. Thus there are four distinct transaction types, abbreviated as follows:

- Outbound Asynchronous (OutAsync)
- Outbound Synchronous (OutSync)

- Inbound Asynchronous (InAsync)
- Inbound Synchronous (InSync)

When the synchronous or asynchronous request handler receives a message, the handler searches the transaction definitions for active transactions which specify that message definition and version. Based on the transactions found, the handler directs the message to the specified nodes.

Transaction Definitions for Cross-Node Messaging

Each Integration Broker node involved in an integration point must define a transaction for each node with which it communicates. For example, for an integration in which node **A** transmits message **X** to both node **B** and node **C:**

- In the node **A** database, an **outbound** transaction specifying request message MSG_X is part of the **NODE_B** definition.
- In the node **A** database, an **outbound** transaction specifying request message MSG_X is part of the **NODE_C** definition.
- In the node **B** database, an **inbound** transaction specifying request message MSG_X is part of the **NODE_A** definition.
- In the node **C** database, an **inbound** transaction specifying request message MSG_X is part of the **NODE_A** definition.

Notice that for this integration, no transactions need to be defined for the default local node in any of these databases.

Note. For a basic integration point, all transactions involved must be of the same transmission type — all synchronous or all asynchronous. However, you can mix transmission types by defining transaction modifiers; see "Administering Relationships."

Transaction Definitions for Local Messaging

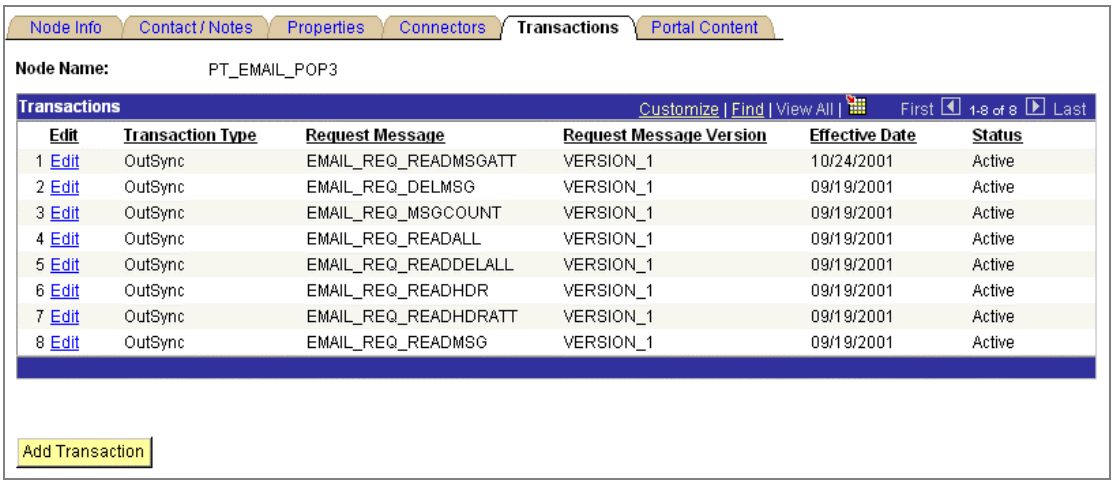
Because a local integration involves only the default local node, it requires only a single transaction. For example, for an integration in which node **A** transmits message **X** to itself (a local transaction):

- In the node **A** database, an **outbound** transaction specifying request message MSG_X is part of the **NODE_A** definition.

  No other transaction is necessary — because NODE_A is the default local node, the inbound transaction is implicit, and the Integration Engine automatically routes it appropriately.

Viewing the Transaction List

Access the Node Definitions – Transactions page.



Node Definitions – Transactions page

Edit Select **Edit** to modify properties of the transaction. The Transaction Detail page appears. See Editing Transaction Details.

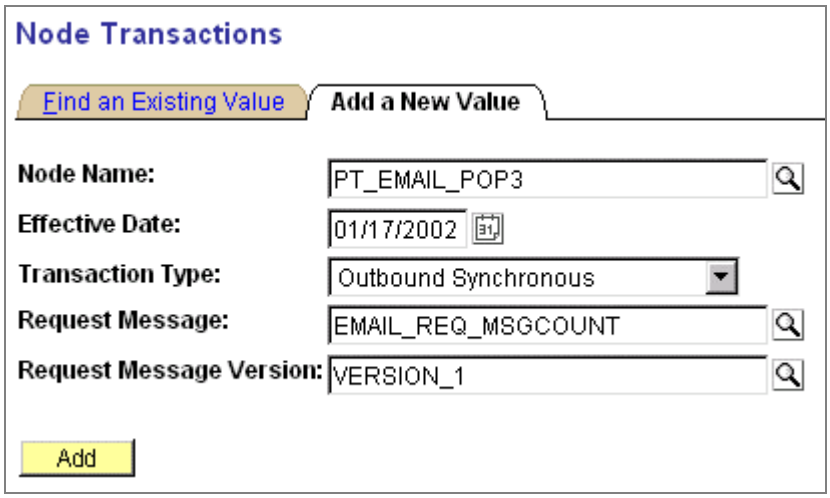
Add Transaction Click to define a new transaction. The Node Transactions page appears. See Defining a Transaction.

This page lists all the transactions associated with the current node, including the transaction type, the request message and version, and the effective date for each transaction.

Note. The transactions listed on this page always define interactions between the current node and the default local node (the database you're signed on to). If the current node *is* the default local node, the transactions are local. The transaction direction is always with respect to the default local node — outbound is *from* the default local node, and inbound is *to* the default local node.

Defining a Transaction

Use the Node Transactions page to specify the basic parameters of a new transaction.



Node Transactions page

Node Name Select the name of a node defined in the current database. The name of the current node is already entered, but you can select a different one with which to associate this transaction.

Note. If you select a different node than the current one, the selected node's definition opens when you save the transaction.

Effective Date Select the date this transaction should go into effect. The current date is the default value.

Transaction Type Select from the following types:

Outbound Asynchronous: The default local node sends a request message to the selected node.

Outbound Synchronous: The default local node sends a request message to the selected node, requiring a response.

Inbound Asynchronous: The default local node receives a request message from the selected node.

Note. This value isn't supported for the default local node definition.

Inbound Synchronous: The default local node receives a request message from the selected node, requiring a response.

Note. This value isn't supported for the default local node definition.

Request Message Select the message to be sent or received with this transaction.

	Note. PeopleCode associated with the message must support the transaction type you selected. See "Sending and Receiving Messages."
Request Message Version	Select the version of the request message to be sent or received with this transaction.

Click the **Add** button to confirm the parameters you entered for the new transaction. The Transaction Detail page appears.

Note. You can edit an existing transaction definition in your application database by accessing the Find an Existing Value tab, where you can search for a transaction based on any of its key parameters. If you select a transaction associated with a different node than the current one, the selected node's definition opens when you save your changes.

No Multiple Synchronous Targets

In practice, only one target node at a time can receive a message sent with a synchronous transaction. Even though you can define the same outbound synchronous transaction for multiple nodes, you must make sure the transaction resolves to a single target node, in one of the the following ways:

- Specify the target node in your sending PeopleCode as a parameter of the Message class SyncRequest method.
- Specify the target node in your sending PeopleCode as a parameter of the SyncRequestXMLDoc built-in function.
- Filter the available target nodes in your OnRouteSend PeopleCode. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

If you specify a target node in your sending PeopleCode, *no* other nodes are considered valid targets, including those with a routing type of *Implicit*.

See Also

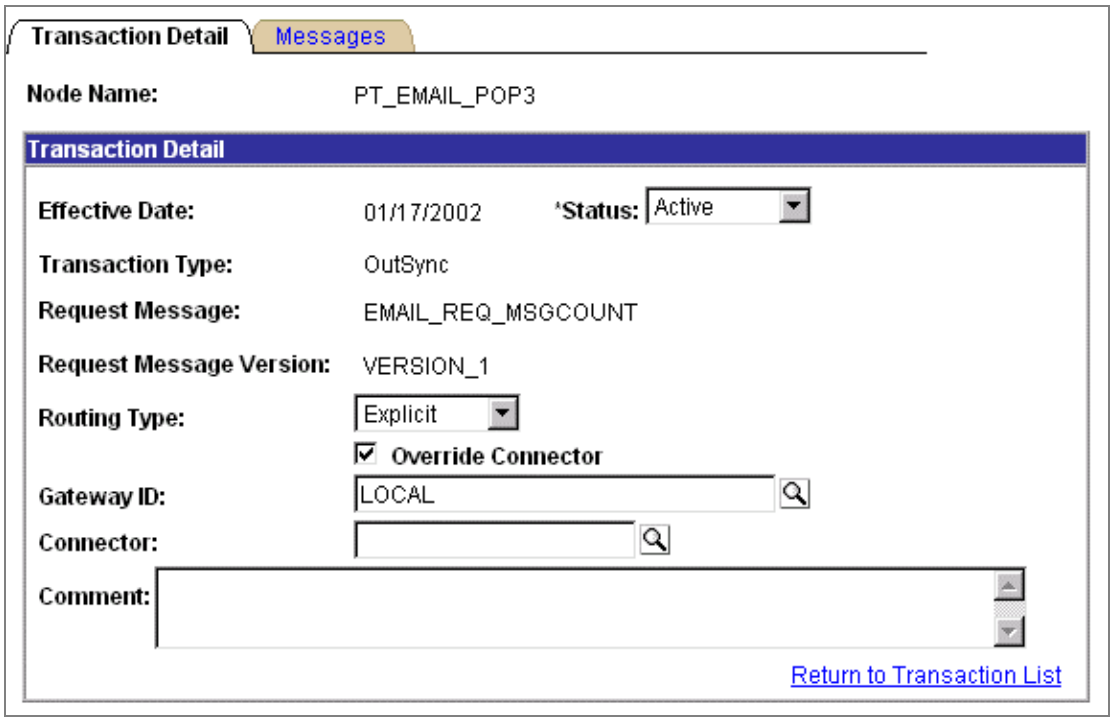
PeopleTools PeopleBook: PeopleCode Reference, "Message Class"

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

Editing Transaction Details

The Transaction Detail page appears when you select Edit on the Node Definitions – Transactions page, or click the Add button on the Node Transactions – Add a New Value page.

Note. The transaction's basic parameters are read-only, and can be altered only by defining a new transaction with different values.



Transaction Detail page

Status Select from the following:

Active: When active, this transaction is considered for use by the request handlers.

Inactive: When inactive, the transaction isn't considered for use by the request handlers.

Routing Type (Outbound transactions only) Select a routing type to override the selected node's specified routing type, which is the default value — for details, see Specifying General Node Information. Select from the following types:

(blank): The selected node's routing type applies.

Explicit: The selected node won't be included as a target node unless you specify it directly in your sending PeopleCode. This overrides the node's specified routing type.

Note. You can also include the selected node by specifying it in the OnRouteSend PeopleCode event associated with the message definition. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Implicit: All nodes with this routing type are included as target nodes unless your sending PeopleCode references specific ones. In that case, only the referenced nodes are targets. This overrides the selected node's specified routing type.

Note. You can filter the list of included nodes using the OnRouteSend PeopleCode event associated with the message definition, to produce a single destination node. You can find more information about this subject on PeopleSoft Customer Connection, under *Integration Broker Advanced Topics.*

Override Connector (Outbound transactions only) If you select this check box, you can choose a gateway and target connector for this transaction that overrides those specified for the selected node. The Gateway ID and Connector fields appear.

Note. If you select this check box, you must select a gateway ID and a connector.

Gateway ID Select the gateway ID for the gateway you want this transaction to use.

Connector Select the connector you want the current transaction to use from the list of connectors registered with the selected gateway. When you save the transaction, a new page appears where you can override the connector's properties. See Editing Transaction Connector Properties.

Return to Transaction List Select Return to Transaction List, and the Transactions page for the selected node appears.

Note. Be sure to save the transaction before returning to the transaction list. If you don't, your changes — including the creation of a new transaction — will be canceled.

Activating Transactions

When a transaction is active, Integration Broker can immediately use it to route messages to or from its associated node. However, messaging also depends on the status of the messages and the node — their definitions in the local database must all be set to active status. The following table outlines how this dependency is enforced for a given message and node:

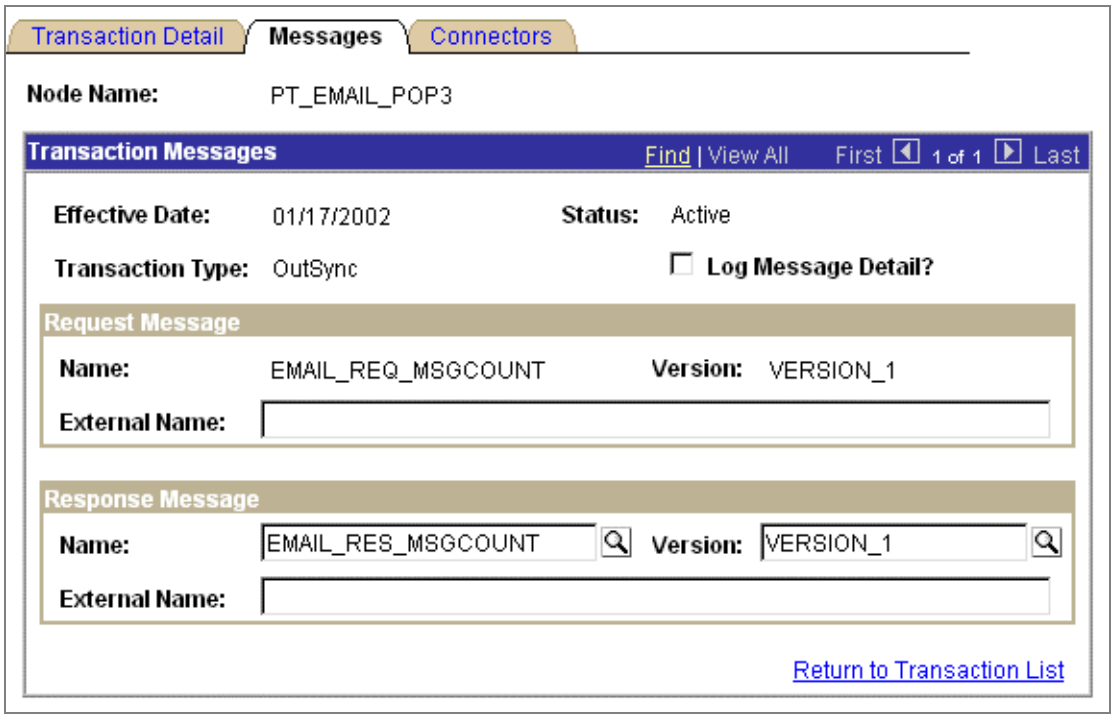
Condition	*Action*	*Dependency*
Message is active	Deactivate the message	All associated transactions are deactivated
Message is active	Create a new transaction	Transaction is active by default
Message is inactive	Create a new transaction	Transaction is inactive by default
Message is inactive	Activate a transaction	Allowed, with a warning

Condition	*Action*	*Dependency*
Message is inactive	Activate the message	No dependency
Node is active	Deactivate the node	All associated transactions are deactivated
Node is active	Create a new transaction	Transaction status matches message status
Node is inactive	Create a new transaction	Transaction is inactive by default
Node is inactive	Activate a transaction	Not allowed
Node is inactive	Activate the node	No dependency
Message and node are active	Activate a transaction	Allowed
Message and node are any status	Deactivate a transaction	No dependency

Editing Transaction Message Details

Access the Messages page from the Transaction Detail page.

Note. The Log Message Detail check box and the Response Message section appear only if you defined this transaction as synchronous.



Messages page

Log Message Detail?	(Synchronous transaction only) Normally, the Integration Broker Monitor logs only header information for synchronous transactions, because the full details can be extensive. Select Log Message Detail if you need to see the full message details for this transaction logged in the Integration Broker Monitor.
External Name	The name of the request or response message as defined in the default local node's database might not match the name by which it's defined in the selected node's database. You can enter an External Name for the message that the selected node will recognize. The external name only applies to the message when it's sent or received by the selected node using this transaction. The defined message name remains valid when the message is being processed by the default local node.
(Response Message) Name	(Synchronous response message only) Select the response message to be sent or received by the selected node. **Note.** This message must be in the same channel as the request message.
(Response Message) Version	(Synchronous response message only) Select the version of the response message that the selected node will use in this transaction.
Return to Transaction List	Select Return to Transaction List, and the Transactions page for the selected node appears. **Note.** Be sure to save the transaction before returning to the transaction list. If you don't, your changes — including the creation of a new transaction — will be canceled.

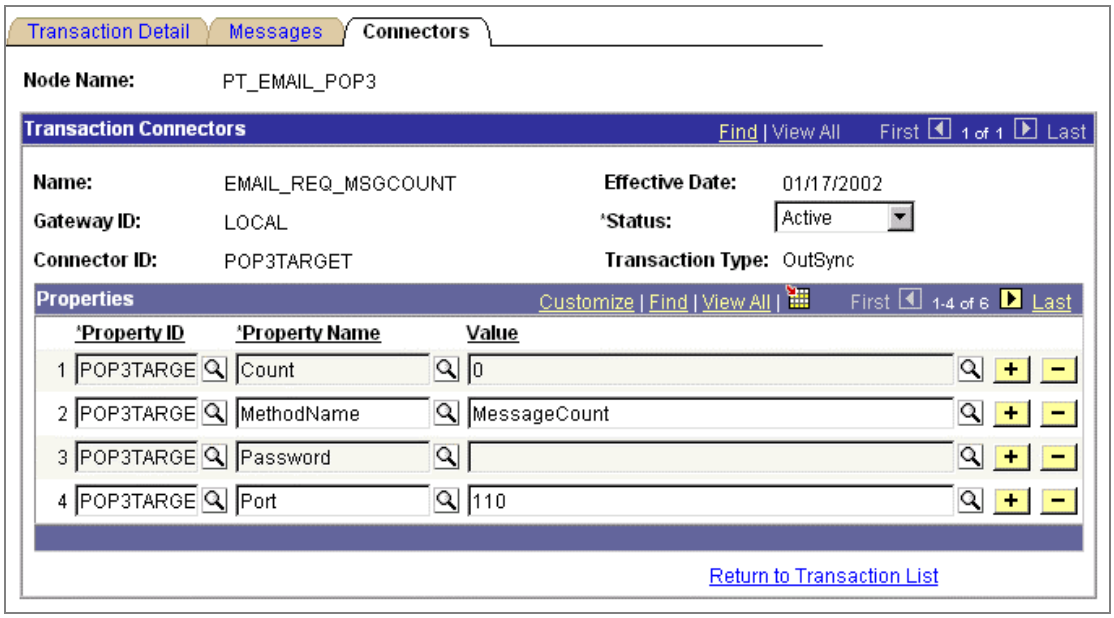
See Also

"Using Integration Broker Monitor"

Editing Transaction Connector Properties

Access the Connectors page from the Transaction Detail page.

Note. This page appears only if you specified an override connector on the Transaction Detail page and saved your changes.



Connectors page

The **Status** field duplicates the Status field on the Transaction Detail page — see Editing Transaction Details.

Because you selected a gateway and target connector for this transaction that overrides those specified for the selected node, you can also override the connector's properties. You modify the **Property ID**, **Property Name** and **Value** fields on this page the same way as the equivalent fields on the Node Definitions - Connectors page — see Specifying a Connector.

Select **Return to Transaction List**, and the Transactions page for the selected node appears.

Note. Be sure to save the transaction before returning to the transaction list. If you don't, your changes — including the creation of a new transaction — will be canceled.

CHAPTER 8

Using Integration Broker Monitor

The **Integration Broker Monitor** is a tool that system administrators use to monitor asynchronous and synchronous message transaction information, status information, and more on the Integration Engine.

The Integration Broker Monitor provides:

- Views of run-time tables, providing status information on channels, nodes, and individual messages. It also provides access to view and, depending on the message status, edit message XML.
- The ability to control and administer all domains that have pub/sub servers running against the current database, including activating or deactivating domains, recovering from stalls, and so forth.
- Workflow notification of messages where errors have occurred as well as archiving functionality.
- Ability to run batch processes for error notification.
- Ability to run batch processes to archive messages for maintenance and increased performance.
- Queries that enable you to gather information about the messaging system meta data, without having to access Application Designer. For example, you can perform queries to retrieve information about the transactions that exist on the system for a given request or response message, the messages that belong to each channel, the channels and messages in the local node, and more.

The Integration Broker Monitor is designed for system administrators, not for end-users.

This chapter discusses:

- Asynchronous messages.
- Synchronous messages.
- Message status information.
- How to open Integration Broker Monitor.
- How to set Integration Broker Monitor security settings.
- How to use the Monitor Message component.

- How to use the Message Details component.
- How to use the Synchronous Details component.
- How to use the Error Notification component.
- How to use the Archive Messages component.
- How to use Message Monitor Component Interface.

Understanding Asynchronous Messages

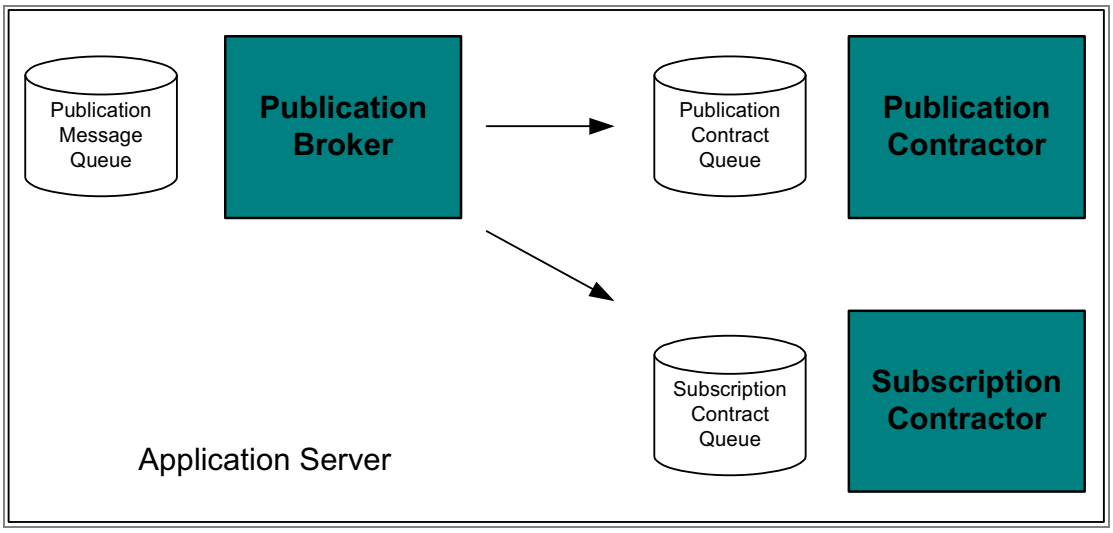
To monitor asynchronous message information in your system, you primarily use the Monitor Message component and the Message Details component in the Integration Broker Monitor. These components enable you to monitor aspects of asynchronous message subscription and publication in your messaging system, such as publication contracts, subscription contracts and more.

This section provides an overview of asynchronous messaging concepts and discusses:

- Brokers, contractors and queues.
- Message server processes.
- Dispatchers and handlers.
- Asynchronous message publication flow and message status information.
- Asynchronous message subscription flow and message status information.

Brokers, Contractors and Queues

The Publication Broker, Publication Contractor, and Subscription Contractor are the primary application server elements required for asynchronous messaging. In terms of a hierarchy, the Publication Contractor and the Subscription Contractor are on the same level below the Publication Broker, which distributes, or routes, the workload to both Contractor server processes.



Brokers and Contractors

Notice that each broker or contractor has its own queue to hold the messages it needs to process. The following sections describe what each service does.

Publication Broker — Acts as the routing mechanism. When an asynchronous message arrives in its queue, the Publication Broker runs the defined routing rules. If the message needs to be published to a remote node it routes the message to the Publication Contractor. If the message is subscribed to on the local node, then the Broker routes the message to the Subscription Contractor. Routing involves submitting either a Subscription or Publication Contract to the appropriate contractor followed by an asynchronous call to the contractor notifying it that it has work in its queue.

Publication Contractor — References the Publication Contract submitted by the Publication Broker, and performs an HTTP post of the publication message to the Integration Gateway. When the Integration Gateway sends a reply indicating that it received the publication message, the Publication Contractor updates the Publication Contract with the status of subscription processing (Done or Retry).

Subscription Contractor — References the Subscription Contract submitted by the Publication Broker, and it executes the appropriate subscription PeopleCode. Then it updates the Subscription Contract concerning the status of the subscription processing.

Message Server Processes

The application server offers six server processes to handle asynchronous messages. They work in pairs to provide three primary services:

Service	*Server Process*	*Description*
Publication Broker	PSBRKDSP	Broker Dispatcher
	PSBRKHND	Broker Handler
Publication Contractor	PSPUBDSP	Publication Dispatcher
	PSPUBHND	Publication Handler
Subscription Contractor	PSSUBDSP	Subscription Dispatcher
	PSSUBHND	Subscription Handler

Dispatchers and Handlers

As mentioned previously, the Publication Broker, Publication Contractor, and Subscription Contractor are each comprised of two separate server processes working together to handle any incoming requests. One server process functions as a "dispatcher" while the other functions as a "handler."

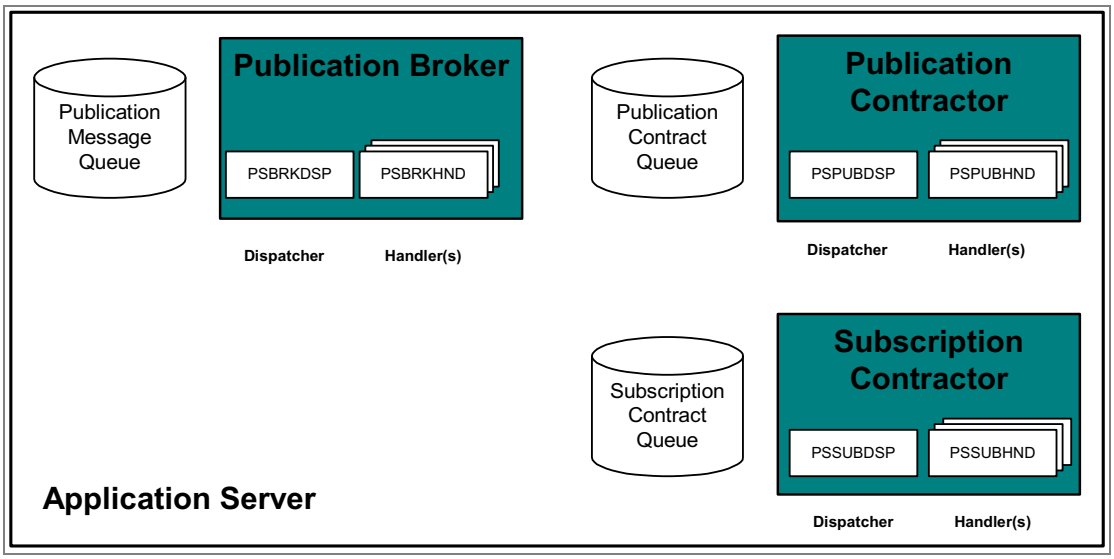
This relationship is analogous to the way that the application server handles workstation connections/requests. To handle the incoming client requests, the application server has a listener and a handler (or a pool of handlers). The listener receives the incoming requests, and then routes them to an available handler.

Typically, you have one listener to many handlers. The relationship between the dispatcher and the handlers is analogous to the relationship between the Jolt Server Listener (JSL) and the Jolt Server Handler (JSH). In the case of the application messaging server processes, think of the dispatcher as the listener, and the handler as similar to the JSH.

So, for the components discussed thus far (Publication Contractor, Subscription Contractor, and Publication Broker) there are *at least* two server processes doing the work: a single "dispatcher" and one or more "handlers". The PSXXXDSP server process is the dispatcher, and the PSXXXHND server process is the handler.

Note. The "XXX" represents BRK, PUB, or SUB. For example, in the case of the Publication Broker, PSBRKDSP is the dispatcher, and PSBRKHND is the handler for the Publication Broker.

The following diagram depicts the messaging server processes grouped within their specific functional position in the messaging architecture.

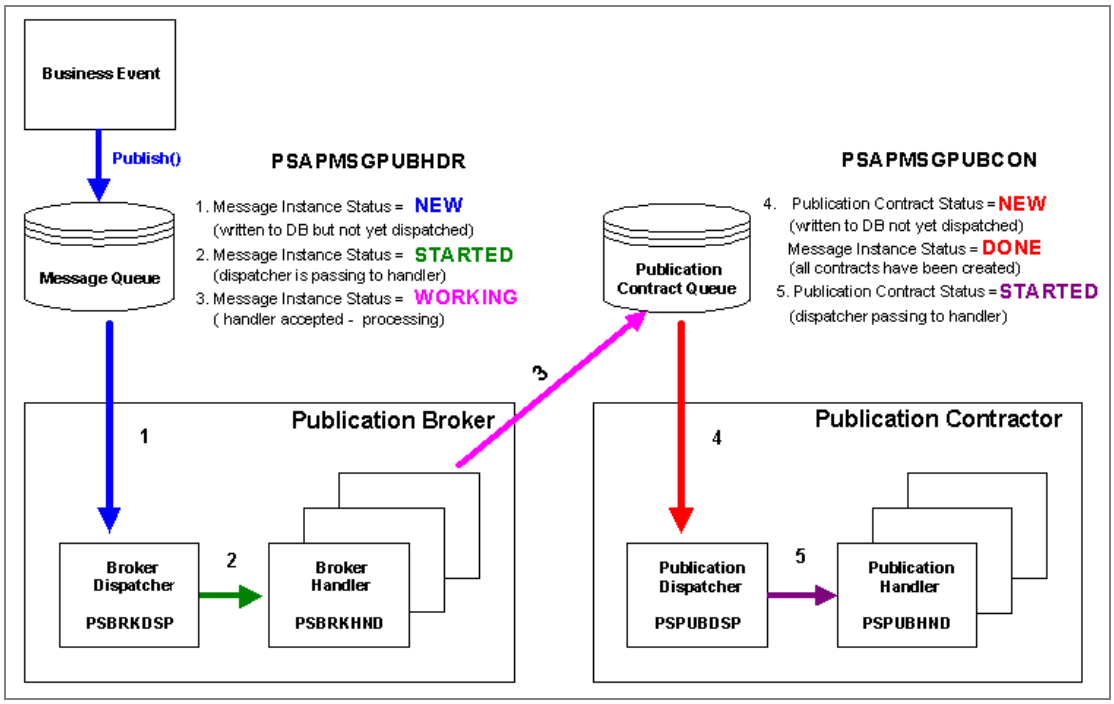


Dispatchers and Handlers

Asynchronous Publication Flow and Message Status

This section describes the flow of an asynchronous message publication through PeopleSoft Integration Broker, from the message instance being published and entering the message queue, to the publication contract creation, and finally to the publication contract being published to the destination node. In addition, the status of the message as it appears in the Integration Broker Monitor is described.

Asynchronous Publish Flow of a Message Instance



Asynchronous Publication Flow of a Message Instance

The processing steps of a publish of an asynchronous message instance are:

1. The message is published and enters the Message Queue. The Broker Dispatcher picks up the message instance from its queue. During this stage the status of the message instance is *New*.

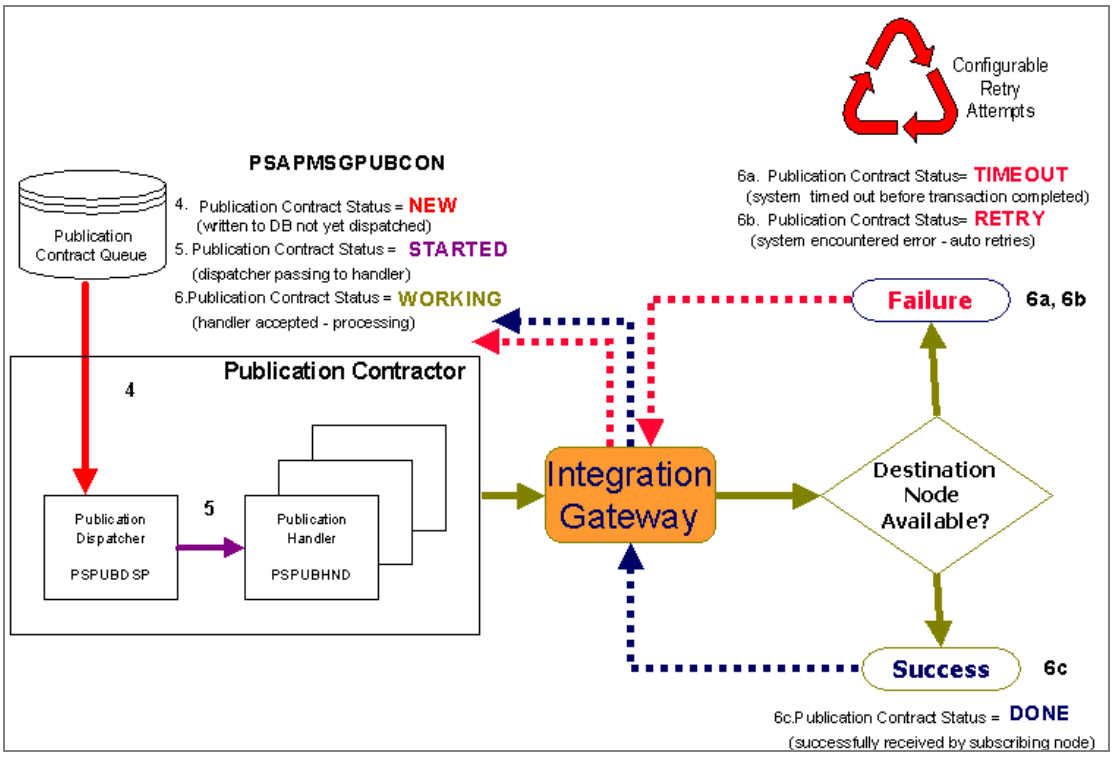
2. The Broker Dispatcher (PSBRKDSP) passes the message instance to the Broker Handler. During this stage the status of the message instance is *Started.*

3. The Broker Handler (PSBRKHND) accepts the message instance, reads the data and runs the routing rules to determine where the publication is to be delivered. The Broker Handler then writes a publication contract in the PSAPMSGPUBCON table (the Publication Contract Queue) and notifies the Publication Contractor that it has an item to process. During this stage the status of the message instance is *Working*.

4. Once the message is stored in the Publication Contact Queue the status of the publication contract is *New*, the message instance status is *Done*, and the Publication Dispatcher picks up the publication contract from its queue.

To view status information for a message instance, select **PeopleTools**, **Monitor Message**.

Asynchronous Publish Flow of a Publication Contract



Asynchronous Publish Flow of a Publication Contract

The previous graphic shows the publish flow of an asynchronous publication contract. The processing steps are:

1. The Publication Dispatcher (PSPUBDSP) passes the publication contract to the Publication Handler. At this stage the status of the publication contract is *Started.*

2. The Publication Handler (PSPUBHND)accepts the publication contract andattempts to deliver the message to the Integration Gateway. At this stage the status of the publication contract is *Working*.

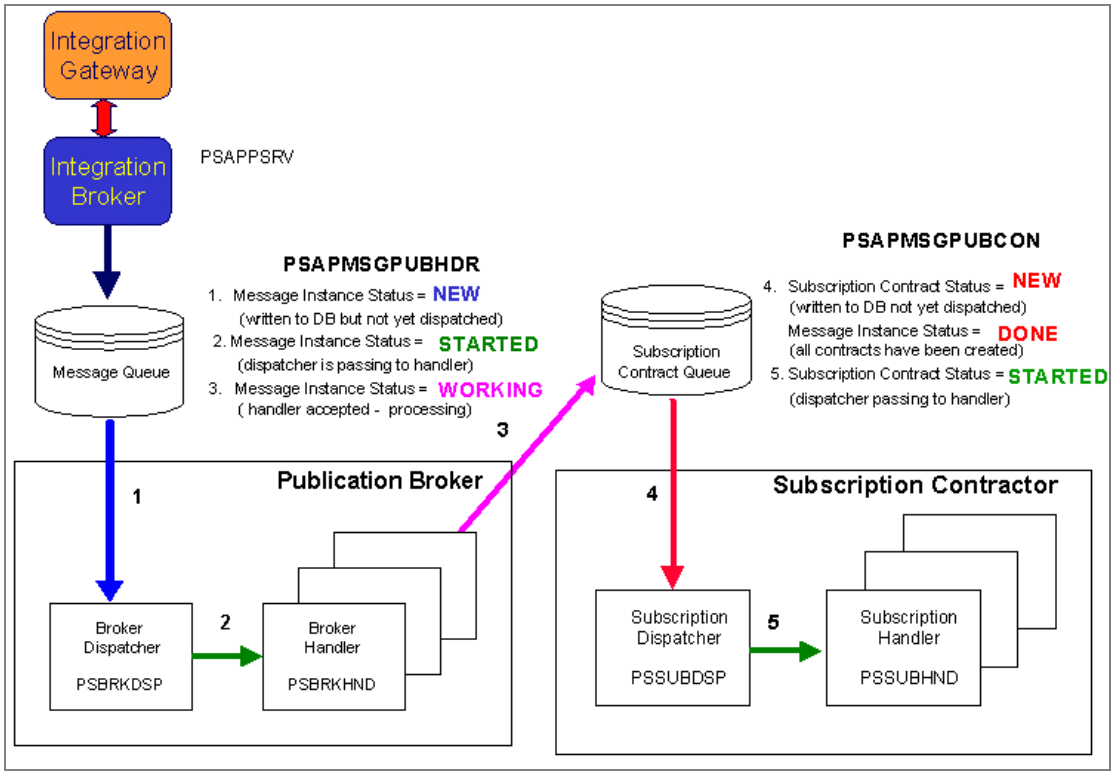
   If the publication contract is successfully delivered to the destination node, the status is *Done* (6c in the diagram*)*. If an error occurs during this stage, the status is *Error*. If the system times out before the transaction is completed, the status is *Timeout* (6a in the diagram). If the delivery fails, the Publication Handler retries the delivery, and the status is *Retry* (6b in the diagram). The message goes back to the Publication Handler and returns to *Working* status.

You can view the status information for the publication contract in Integration Broker Monitor by selecting **PeopleTools**, **Monitor Message**, **Pub Contracts**.

Asynchronous Subscription Flow and Message Status

This section describes the flow of an asynchronous message subscription through Integration Broker, from the message instance entering the message queue, to the subscription contract creation, and finally to the subscription PeopleCode program processing the message and updating the application data tables. In addition, the status of the message is described at each stage as well.

Asynchronous Subscription Flow of a Message Instance



Asynchronous Subscription Flow of a Message Instance

The graphic shows the subscription flow of an asynchronous message instance. The processing steps are:

1. The message enters the Message Queue. The Broker Dispatcher picks up the message instance from its queue. During this stage the status of the message instance is *New*.

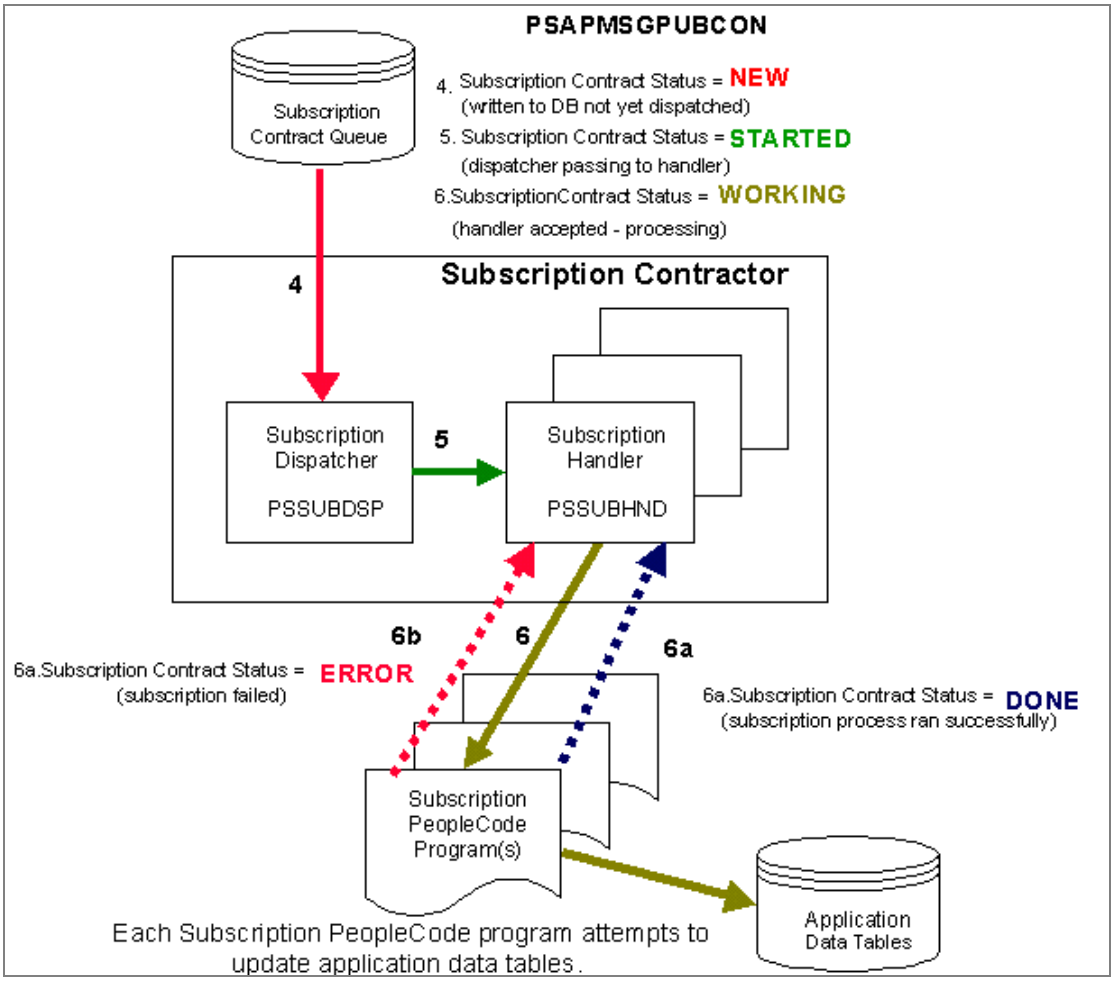
2. The Broker Dispatcher (PSBRKDSP) passes the message instance to the Broker Handler. During this stage the status of the message instance is *Started*.

3. The Broker Handler (PSBRKHND) accepts the message instance, reads the data and runs the subscription routing rules to if the message is to be processed locally. The Broker Handler then writes a subscription contract in the PSAPMSGPUBCON table (the Subscription Contract Queue) and notifies the Subscription Contractor that it has an item to process. During this stage the status of the message instance is *Working*.

4. Once the message is stored in the Subscription Contact Queue the status of the subscription contract is *New*, the message instance status is *Done* and the Subscription Dispatcher picks up the subscription contract from its queue

You view the message instance status in the Message Instance page of the Monitor Message component. In this example, at the point the status of the asynchronous message instance is *Done*, the subscription contract status is *New*.

Asynchronous Subscription Flow of a Subscription Contract



Asynchronous Subscription Flow of a Subscription Contract

The diagram shows the flow of what is now the subscription contract through the system, including its status at each stage of the flow. The processing steps are:

1. The Subscription Dispatcher (PSSUBDSP) passes the subscription contract to the Subscription Handler. At this stage the status of the subscription contract is *Started*.

2. The Subscription Handler (PSSUBHND) accepts the subscription contract and executes the subscription PeopleCode. In the example shown in the diagram, the subscription PeopleCode then uses the message data to update application data tables. However, the

subscription PeopleCode can use the message data as input to look up information, create and publish another message, and so forth. At this stage the status of the publication contract is *Working*.

If the subscription PeopleCode runs successfully, the status is *Done* (6a in the diagram). If an error occurs during this stage, the status is *Error* (6b in the diagram).

To view status information for subscription contracts, select **PeopleTools**, **Monitor Message**, **Sub Contracts**.

Understanding Synchronous Messages

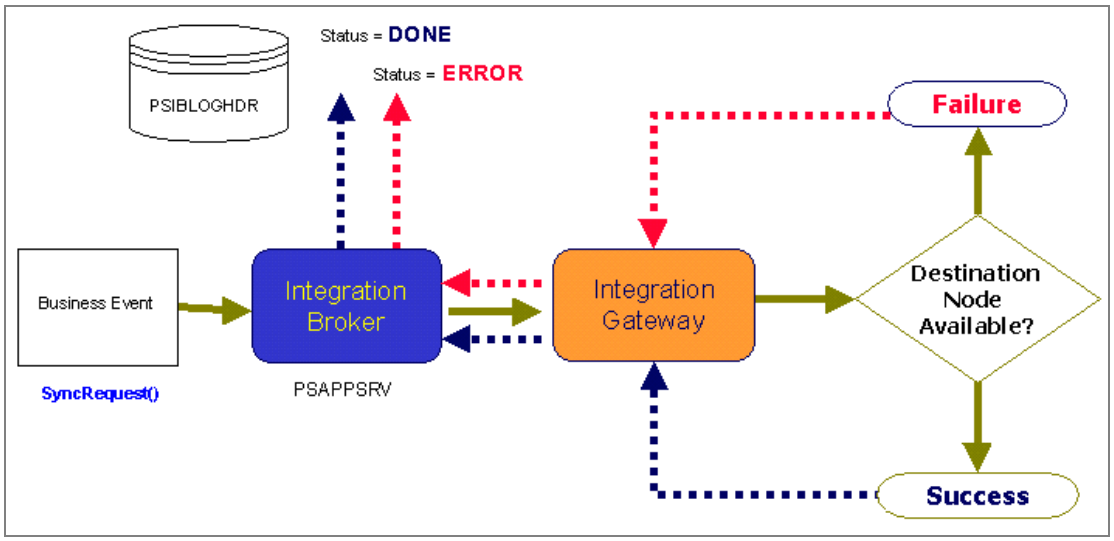
To monitor synchronous message information in your system, you primarily use the Monitor Message component and the Synchronous Details component in the Integration Broker Monitor. These components enable you to monitor aspects of synchronous message subscription and publication, such as node status, channel status, domain status and more.

This section discusses:

- Synchronous message publication flow and message status information.
- Synchronous message subscription flow and message status information.

Synchronous Publication Flow and Message Status

The following diagram illustrates synchronous message publishing through Integration Broker and the corresponding message status at each stage.



Synchronous Publication Flow of a Publication Contract

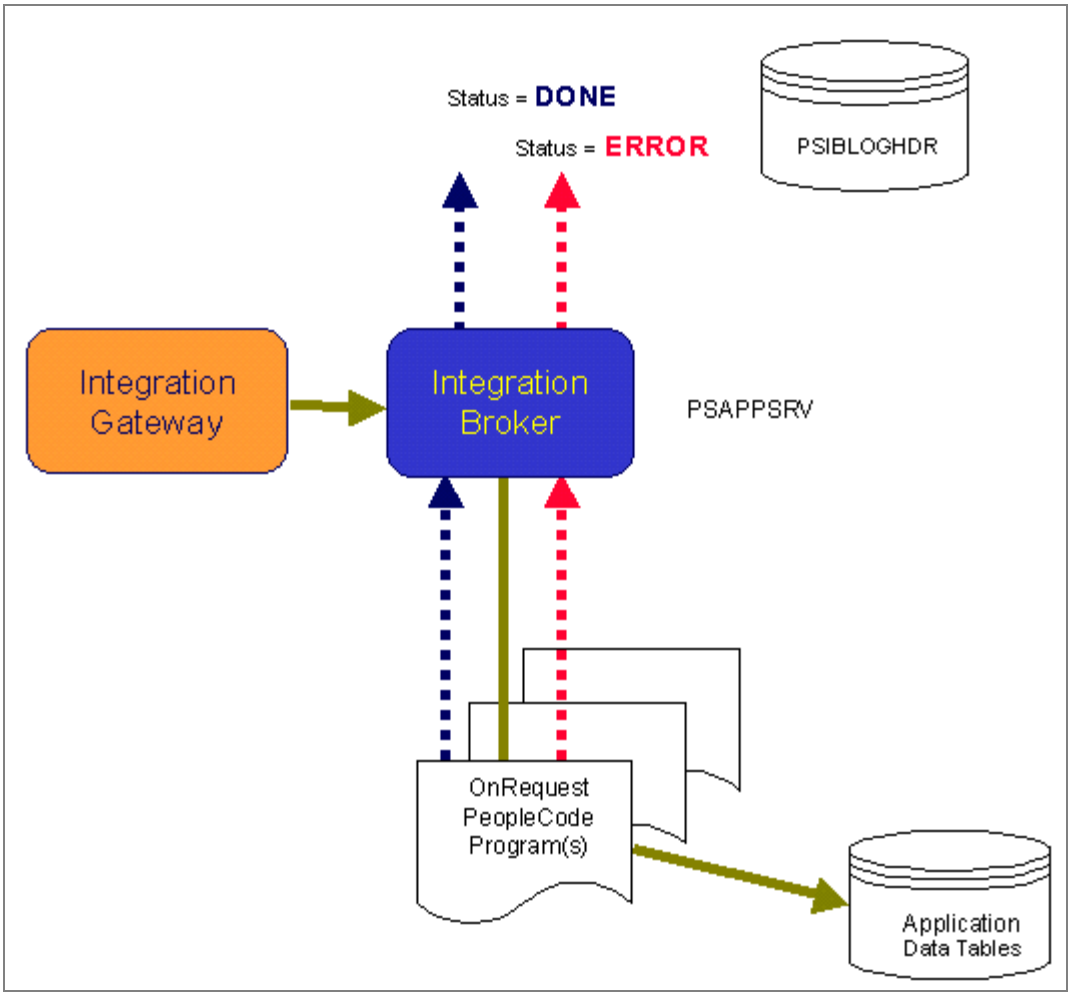
The processing steps of a synchronous publication contract through Integration Broker are:

1. The systems makes a SyncRequest ( ) call to the Integration Broker.
2. Integration Broker sends the message to the Integration Gateway.
3. If the Integration Gateway is able to deliver the message to the destination node the process is successful and the status is *Done*. If the process in unsuccessful, the status is *Error*.

You can view the status information for the publication contract in Integration Broker Monitor by selecting **PeopleTools**, **Monitor Message**, **Synchronous Messages**.

Synchronous Subscription Flow and Message Status

The following diagram illustrates synchronous message subscription through Integration Broker, and the corresponding message statuses at each stage.



Synchronous Subscription Flow of a Subscription Contract

The previous diagram shows the flow of a synchronous subscription contract through the system. The processing steps are:

1. The Integration Gateway passes an inbound synchronous message to the Integration Engine.
2. The Integration Engine executes a subscription PeopleCode program.
3. The subscription PeopleCode program runs and attempts to update the application data tables. If the program runs successfully, the status is *Done*. If the subscription PeopleCode program fails, the status is *Error*.

You can view the status information for the publication contract in Integration Broker Monitor by selecting **PeopleTools**, **Monitor Message**, **Synchronous Messages**.

Getting Started Using Integration Broker Monitor

This section discusses how to:

- Open Integration Broker Monitor.
- Set Integration Broker Monitor security and permissions.

Opening Integration Broker Monitor

To open Integration Broker Monitor, select **PeopleTools, Integration Broker Monitor, Monitor**, and select a component with which to work.

There are five components associated with the Integration Broker Monitor, as described in the following table.

Monitor Messages	View information about message instances, publication contracts, subscription contracts, synchronous messages, channel status, node status and domain status. Generate a number of related pre-defined queries.
Message Details	View asynchronous message details, including information about the message instance, its publication or subscription contracts, error messages and message instance XML. If transformations have been applied to the message, you can view the transformed XML for the publication and subscription contracts.
Synchronous Details	View information about synchronous message details, message errors and view request and response XML, before or after transformation.
Error Notification	Run batch processes to notify you of issues affecting your messaging system.
Archive Messages	Run the batch process to archive messages.

Setting Security and Permissions

Before using the Integration Broker Monitor, you must set up security for it using permission lists. Make sure that the individuals who need access to it have the proper authority. To navigate to permission lists, select **PeopleTools**, **Maintain Security**, **Use, Permission Lists**.

You use permission lists to grant access to the Integration Broker Monitor pages and also to grant read and/or write access to specific message channels. Users can only view messages in channels to which they have read access, and any edit action requires write access.

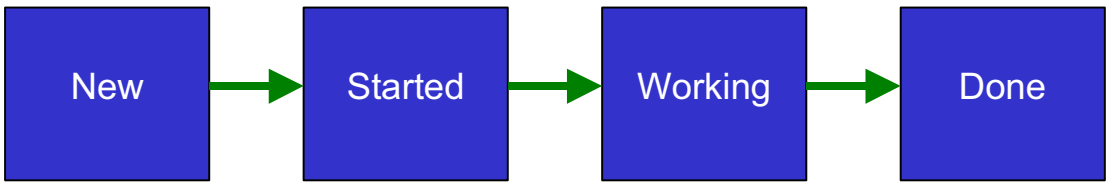
Understanding Message Status Information, Blocked Channels and Stalled Channels

This section discusses:

- Message status information.
- Blocked channels.
- Stalled channels.

Message Status Information

The Integration Broker Monitor displays the following status information for asynchronous message during the normal progression of them through the system.



Synchronous messages can have only two statuses, *Done* and *Error*.

However, if errors or other situations arise, Integration Broker Monitor can display different status information for messages.

Note. Synchronous messages can have only two statuses, *Done* and *Error*.

New	Either the item has been written to the database, but has not been dispatched yet, or the item has just been resubmitted.
Started	The dispatcher is in the process of passing the item to a handler, but the handler has not received it yet.
Working	The handler has accepted the item and is currently processing it.

Done	For asynchronous messages, the handler successfully processed the item. For synchronous messages, this status indicates different outcomes, depending on the type of process you are monitoring. • **Message Instance**. All contracts have been created. • **Publication Contracts**. The publication has been successfully received by the subscribing node • **Subscription Contracts**. The subscription process ran successfully.
Error	An error occurred during processing. Manual intervention is required.
Retry	The system encountered an intermittent error during processing. The system retries messages with this status automatically.
Timeout	The system has reached the maximum retry count to send a message.
Edited	The publication data for the item has been edited. Processing will not resume until you resubmit the item.
Cancelled	The item has been cancelled. The system cannot process the item until you resubmit. it

Blocked Channels

Blocked channels are by design to preserve the order in which messages get processed.

The Pub/Sub system guarantees that items are processed in the order it receives them. If a message has a status of *Error*, *Timeout* or *Edited,* the message queue becomes blocked and no processing will continue until you resolve the issue with the message.

Publications, Publication Contracts and Blocked Channels

For publications, channels are partitioned in queues by sub-channels.

For publication contracts, the channel is further partitioned into queues by sub-channel and target node. If a channel is ordered, items in that channel and in the same queue are processed in the order received. The dispatcher does not begin processing an item until all items ahead of it in the queue are *Done* or *Cancelled*. An item with a status of *Error*, *Timeout*, or *Edited* will block all items behind it in the same queue. If the remote node is unavailable the dispatcher does not attempt to process the contract, and the queue is blocked until the remote node becomes available, which is why publication contracts are partitioned by target node.

If, however, a channel is unordered, one item (that is the publication, publication contract, or subscription contract) never block another item—all items are processed in parallel.

Stalled Channels

Stalls are not by design. They are caused by gaps in functionality, user errors, defects and so forth.

For example, a channel can become stalled when:

- Multiple domains are accessing the same database and one of the domains is shutdown abnormally, and items are left in *Started* or *Working* status. (Note that you can use the Domain Status page to recover from a situation like this.)
- Changing the Pub/Sub runtime tables through direct SQL. Dispatcher "in-memory" copies of the database tables do not get updated. In this situation, you must reboot the dispatchers.

Monitoring Messaging System Information

You can monitor message system queue information using the Monitor Message component. The Monitor Message component enables you to view information about message instances, publication contracts, subscription contracts, synchronous messages, channel status, node status and domain status. You can also use this component to generate related pre-defined queries.

This section discusses how use the Monitor Message component to:

- Monitor system queue information.
- Monitor asynchronous message instances.
- Monitor publication contracts.
- Monitor subscription contracts.
- Monitor synchronous message instances.
- Monitor channel status.
- Work with node status.
- Work with pub/sub server domains.
- Run message queries.

Working with the Monitor Messages Component

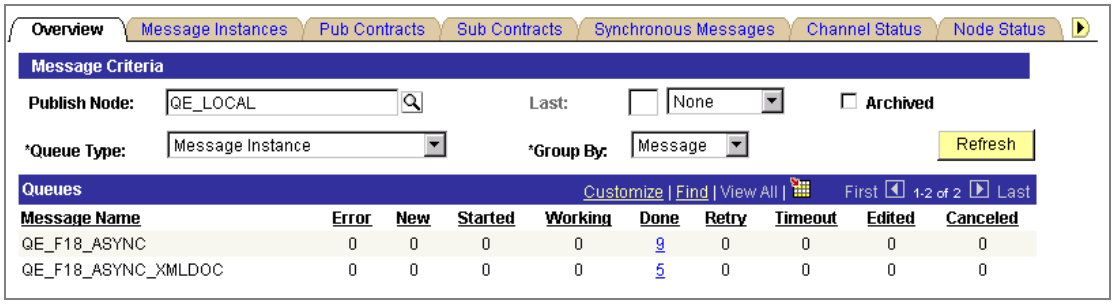
This section discusses how to:

- Open the Monitor Messages component
- Use the Monitor Message component pages

- Filter messaging information
- Save filtering selections
- View Integration Broker Monitor output

Opening the Monitor Messages Component

To open the Monitor Message component, select **PeopleTools, Integration Broker Monitor, Monitor**, **Monitor Messages**. By default the Overview page is active.



PeopleTools, Integration Broker, Monitor, Monitor Message

Monitor Message Component Pages

The Monitor Message component consists of the following pages:

Overview	Provides a high-level view of your system queues so that you can isolate particular areas and then drill down for further information.
Message Instance	Displays all asynchronous messages from remote nodes or applications that publish information.
Pub Contracts	Shows outbound publication contracts to send to remote message nodes with which the system is interacting.
Sub Contracts	Displays work orders to run PeopleCode programs to which the local node subscribes.
Synchronous Messages	Displays inbound synchronous messages from remote nodes or applications that publish information.
Channel Status	Displays the status of the message channels defined in the system.
Node Status	Displays the status of the local message node. It also allows pinging remote nodes to determine their availability.
Domain Status	Displays the domains with Pub/Sub servers that are running against the application database of the local node. Enables you to gracefully pause or take a domain offline.

	Also enables you to force a reset, which causes the Dispatcher Status for all processes on all machines into cleanup mode.
Queries	Provides a pre-defined set of queries (built with PeopleSoft Query) that enable you to view details and generate reports about the message setup, message channels and message nodes in your system.

Filtering Messaging Information

Before you begin monitoring the numerous facets of your messaging system, there are a few general guidelines to follow to make sure that you quickly drill down to the information you need.

Because the Integration Broker Monitor provides information from every aspect of your messaging system, you need to understand how to filter the information to reduce the number of items before you. For instance, rather than sifting through every message in the entire system, the Integration Broker Monitor enables you to sort by publish node, publish date/time, live and archived messages, and so on.

You can filter the "result set" on the following pages in the Monitor Message component:

- Overview
- Message Instances
- Pub Contracts
- Sub Contracts
- Synchronous Messages

Integration Broker Monitor has general filtering options that apply to each page where filtering applies. The following table provides a description of each general filtering option. Other filtering options may be present in the Integration Broker Monitor and are described where they appear.

Publish Node	Every message is stamped with the node that publishes it. This control provides a drop-down list containing all the nodes defined in your system. The Integration Broker Monitor only allows you to view information for the local system (database). But the local database may have in its queues messages published not only by the local node, but also remote nodes. There is only one local node for a database.
Last	Every message is stamped with the node that publishes it. This control provides a drop-down list containing all the nodes defined in your system. The Integration Broker Monitor only allows you to view information for the local system (database). But the local database may have in its queues messages published not only by the local node, but

	also remote nodes. There is only one local node for a database.
Archived	The Archived box is a toggle switch that enables you to specify a search your archived or "live" message data. To search archived message data, select the checkbox. To search "live" message data, deselect the checkbox.
Channel Name	To view message instances within a specific channel, select the appropriate channel value from the Channel Name dropdown list.
Message Name	To view the instances of a particular message definition, select the appropriate message value from the Message Name dropdown list.
Status	To view message instances by status, select the status criteria from the Status dropdown list. The status options reflect the status columns that appear on the Overview page.

On the pages where filtering applies, you enter your filtering criteria in the Message Criteria section, and the output, or result set, from your criteria appears in the status grid directly below the filtering options.

Filtering options that apply to specific pages are discussed later. For example, the description for filtering options that are specific to viewing Overview information, appear within the discussion of the Overview page.

Saving Filtering Selections

Typically, you look for information on numerous occasions using the same filtering criteria. Often, it is just a matter of becoming familiar with the output and display of information when using a particular set of filtering criteria. Rather than having to reset filtering options each time you launch the Integration Broker Monitor, the system saves your filtering options so that the next time you use Integration Broker Monitor, your previous filtering choices are set automatically.

To save filtering selections

1. Select the filtering options you desire on each page on which filtering applies (Overview, Message Instances, Pub Contracts, and Sub Contracts).

2. After you have selected the appropriate filtering options, click **Refresh** on each page.

   Clicking Refresh not only refreshes the page according to the most recent filtering selections; it also saves the most recent filtering selections to the database. The system then associates a given set of filtering selections with your User ID, and the next time you log in and launch the Integration Broker Monitor the system displays the message data according to your most recent filtering selections.

Viewing Output

The output in the status grids of the Monitor Message component enables you to view information by channel, date, status, and so on.

To sort the information that appears in the status grids, click on a column heading.

After you locate a particular message status, publication contract, subscription contract or other item of interest, click the **Details** link to view detailed data associated with the item. The Details link automatically launches the Message Details component, Synchronous Details component, or a custom, application-specific page where you can view additional data and correct errors.

Whether you use the Message Details component, Synchronous Details or a custom page to view and edit a message depends on the properties the developer selected for the Message Definition in Application Designer. The Details link invokes a Transfer Page to the component specified in the Message Definition, Message Properties, Error Viewing/Correction option.

In addition, many pages of Monitor Message component provide the following control that enables you to view and work with output data.

Click the **Download** button to download output to a spreadsheet.

Monitoring System Queue Information

The Overview page is the first page you see when you access the Integration Broker Monitor. You check this page to get a high-level overview of the status of asynchronous Message Instances, Publication Contracts and Subscription Contracts.

Depending on the information your gather, you then drill down further into the Message Instances, Pub Contracts, and Sub Contracts pages.

Most of the time the status for a message or channel that displays in the Queue section is set to *Done*. This means that the message instance arrived in the Publication Queue (creating the message headers only). However, other statuses for a message or channel can display. For instance, if the Pub/Sub system is down, the status is *New*. If there are transformation or OnRoute PeopleCode errors, the message status is *Error*. In addition, if you were to bring the monitor up at just the right moment, you might see a message status of *Started* or *Working*. Use the other pages in this component to view more comprehensive message status information.

In addition to the filtering options described in Filtering Messaging Information at the beginning of this chapter, you can filter the information displayed on this page using the following criteria.

Queue Type Specify the particular queue in which you are interested in monitoring. Your options are Message Instance, Publication Contract, or Subscription Contract.

Group By	Use this option to view by Channel or Message. After clicking refresh, notice that the label of the following section changes to reflect the option you selected from this drop-down list.

After you select your filtering options, click **Refresh** so that the display reflects your Message Criteria.

See Also

Working with the Monitor Messages Component

Monitoring Asynchronous Message Instances

The Message Instances page enables you to monitor the status and details related to the individual asynchronous message instances that exist in either your live or archived system. Every publication and subscription contract is associated with a message instance.

After you select your filtering options, click **Refresh** so that the display reflects the criteria you selected.

Note. All asynchronous transactions are sent to the Publication Queue regardless of whether the transactions are active or inactive. As a result inactive transactions that display in the Message Instance tab will not contain any publication contracts.

See Also

Working with the Monitor Messages Component

Monitoring Publication Contracts

Shows outbound publication work orders to send to remote message nodes with which the system is interacting. The system does not create Publication Contracts for routing to the local node.

After you select your filtering options, click **Refresh** so that the display reflects your Message Criteria.

In addition, the Pub Contracts page features a **Transaction Retry Queue** link that opens the **Undelivered Node Transaction** window. The Undelivered Node Transaction window provides read-only access to information about undelivered node transactions, such as the message node name, transaction type, request message, and request message version. This information is stored in the Nodes Down table.

The **Force Retry All** button on the Undelivered Node Transaction window allows you to clear the Nodes Down table so the system can attempt to restart any nodes that are down.

For example, if a node is in the Nodes Down table and you change the URL of the node, the node will never free up because it is still considered "down" based on the old URL. By using the Force Retry All button, the system will retry to start the node.

See Also

Working with the Monitor Messages Component

Monitoring Subscription Contracts

The Sub Contracts page enables you to view work orders to run PeopleCode programs to which the local node subscribes. The display does not reveal subscription contracts for remote nodes.

After you select your filtering options, click **Refresh** so that the display reflects your message criteria.

See Also

Working with the Monitor Messages Component

Monitoring Synchronous Message Instances

The Synchronous Messages page displays inbound synchronous messages from remote nodes or applications that publish information.

After you select your filtering options, click **Refresh** so that the display reflects your Message Criteria.

See Also

Working with the Monitor Messages Component

Monitoring Channel Status

The Channel Status page displays the status of the channels defined in the local node. A channel has two states: Running and Paused. If you notice a backup within a particular channel, first you should check the Channel Status page to make sure the channel is running.

There are occasions where you need to shut down a particular node to perform a specific maintenance task. Typically, a node contains numerous channels. The ability to shut down a particular channel as needed means that the other channels on the node continue processing messages undisturbed.

To determine the status of a particular channel, locate it within the Channel Name list, and then check the corresponding value in the Status column.

Working with Node Status

The Node Status page enables you to view, add and delete pause times for local nodes. In addition, you can also use this age to test and ping local nodes.

This section discusses how to:

- Add a pause time to a local node.
- Test local nodes.
- Ping remote nodes.
- View undelivered node transactions.

Scheduled System Pause Times for Local Nodes

A **pause time** refers to an interval of time in which the message node becomes inactive. When the pause time begins, the messaging node is effectively shut down until the pause time is scheduled to end.

There are times when you need to schedule a pause time so that you can perform regular maintenance tasks or devote server resources to an important batch run. For example, say that you have a complex batch program that runs on the same server machine as a particular message node every Monday morning from 12:05 AM to 3:30 AM. To make sure that the batch program has enough memory devoted to it you can set a pause time for the message node that runs from 12:00 AM to 4:00 AM. A pause time like this enables you to make sure batch run has ample system resources to complete successfully within the desired batch window.

During a pause time, messages are not published or received by the local system. When your system is ‘Paused’, the node cannot accept the message sent to it, and the publishing node must attempt to send the message again later. The publishing node continues to send the message until it exceeds the local ‘Time Out Period.' When this happens, the message assumes a ‘Timeout’ status in the publisher’s message queue. Keep in mind that ‘Time Out Period’ is an attribute of the publication channel, not the subscription channel.

If your system attempts to send a message while the message node is paused, the system writes the message to the Pub/Sub queues, but the system cannot physically publish the message until the system is no longer in the paused state.

Note. Pause times do not appear in an Application Designer upgrade project; you cannot upgrade them.

Adding Pause Times to Local Nodes

This section describes how to add a pause time to a local node.

To add a node pause time

1. Click **Add Pause.**
2. Select a day of the week from the **Start Day** dropdown list.

   This value must reflect the day you want the pause to start.
3. Enter a value in the **Start Time** edit box.

   The value you enter here applies to your Start Day. Enter the time of day that you want the pause time to begin.
4. Select a day of the week from the **End Day** dropdown list.

   This value must reflect the day you want the pause to start.
5. Enter a value in the **End Time** edit box.

   The value you enter here applies to your End Day. Enter the time of day that you want the pause time to end and normal processing to resume.
6. After you have entered the appropriate start and end values to define your pause interval, click **OK.**

Deleting Pause Times

The information in this section describes how to delete a pause time.

To delete an existing pause time

1. In the pause time list, locate the pause time (interval) that you want to delete.
2. Click the **Delete** button to the right of the entry in the pause time list.

Testing the Local Node

The information in this section describes how to test the local node to determine if it is active.

To test the local node

1. Make sure you are logged onto the node that you want to test.
2. Click the **Test Node** button.

   Depending on the status of the node, you receive one of the following messages:

```
Node is paused.
```

   Or

```
Node is not paused.
```

Pinging Remote Nodes

To check to see if a remote node is currently running and able to receive messages from the local node, you ping the remote node from the Node Status page. A successful ping indicates that the remote node is currently available. An unsuccessful ping could indicate that the node and/or the gateway are not running.

To ping a remote node

1. In the **Ping a Node to Determine Availability** section, select the node from the **Message Node Name** dropdown list to display a list of active nodes.

   The **Location** column in the grid below reveals the locations defined for the node.

2. Click the **Ping Node** button.

   The **Node Information Section** displays connector information defined for the node.

Viewing Undelivered Node Transactions

The Node Status page also features a **Transaction Retry Queue** link that opens the **Undelivered Node Transaction** window that provides read-only access to information about undelivered node transactions, such as the message node name, transaction type, request message, and request message version. This information is stored in the Nodes Down table.

The **Force Retry All** button on the Undelivered Node Transaction window allows you to clear the Nodes Down table so the system can attempt to restart any nodes that are down.

For example, if a node is in the Nodes Down table and you change the URL of the node, the node will never free up because it is still considered "down" based on the old URL. By using the Force Retry All button, the system will retry to start the node.

Working with Pub/Sub Server Domains

The Integration Broker includes a set of tuxedo servers that monitor database tables and perform processing for items in the tables. The processing can include running PeopleCode programs, creating publication and subscription contracts, and so forth.

The Domain Status page enables you to view the domains that have Pub/Sub servers on them that are running against the application database. You can also use this page to manually set domain grace periods to allow processing to on a domain to complete before you pause processing or take a domain offline.

In addition, if a machine with a domain on it crashes, the messaging system may still expect that processes on the domain are still working on items in the runtime tables. The Domain Status page enables you to set the domains to inactive so that other Pub/Sub servers can complete the processing of these items, thus allowing you to recover from domain and machine crashes.

This section discusses how to:

- Work with the Domain Status page.
- Inactivate Pub/Sub server domains.
- Change dispatcher status for processes.
- Set domain grace periods.

Working with the Domain Status Page

The Domain Status page features three sections, the Domain Criteria section, the Domain Status section and the Dispatcher Status section.

The **Domain Criteria section** enables you to perform actions on all domains on the messaging system, such as apply a grace period all domains, activate or inactivate all domains, and purge the currently displayed information in the Dispatcher Status section.

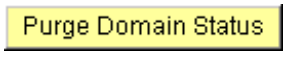
The **Domain Status section** provides application server name and path information for all machines that have domains on the messaging system. For any machine, you can use the dropdown list to activate or inactivate the machine and all domains on it. You can also set grace periods for domains on specific machines.

The **Dispatcher Status section** displays information about machines in the messaging system that have dispatcher processes associated with them. This area displays the machine name, the dispatcher process name, the application server path, the Dispatcher Status and any grace periods set for a process running on the domain.

There are three valid Dispatcher Status values.

ACT	Indicates that the dispatcher process is active on the domain.
INACT	Indicates that the dispatcher process is inactive on the domain. No processing occurs.
CLNUP	Indicates that the dispatcher process is in clean up mode. The Pub/Sub server releases items in queue for processing and waits for items currently processing to finish. The time that displays in the Grace Period column indicates when the cleanup process will end. The time equals the system time and the clean up time interval you enter.

The Domain Status page also features the following buttons:

	Click the **Purge Domain Status** button to purge all of the currently displayed status information in the Dispatcher Status section. After using this feature, information about all processes that are still running automatically populate this section.

Update	Click the **Update** button to saves or apply changes you make in the Domain Criteria section or the Domain Status section.
Force Reset	Click the **Force Reset** button to reset the status of all entries in the Dispatcher Status column in the Dispatcher Status section to *Inactive*.
Refresh	Click the **Refresh** button to refresh the Domains section and Dispatcher Status section of the page.

Inactivating Pub/Sub Server Domains

The section describes how to inactivate Pub/Sub server domains on all machines in your messaging system, or for domains on specific machines.

To inactivate Pub/Sub servers on domains

1. Choose PeopleTools, Integration Broker, Monitor, Monitor Message.
2. Click the **Domain Status** page.
3. Inactivate Pub/Sub server domain(s):
    a. To inactivate domains on all machines in the messaging system, in the **Domain Criteria section** check the **All Domains Inactive.** To activate the servers at a later time, check the **All Domains Active Box**.
    b. To inactivate domains on individual machines, in the **Domain Status section** locate the domain(s) you wish to inactivate. From the dropdown list, select **Inactivate**. To activate the servers at a later time, select **Activate** from the list.
4. Click **Update** to apply the changes.

Note that in the Domain Status section, the Domain Status for the domains you inactivated changed from Active to Inactive. In addition, in the Dispatcher Status section, the Dispatcher Status of all processes associated with the domains is changed to from active to cleanup.

If you inactivated all domains, a **Force Reset** button appears under the Update button. Force Reset enables you to force the Dispatcher Status from cleanup to inactive.

Changing Dispatcher Status for Processes

You can force the system to change the dispatcher status for processes using the **Force Reset** button. This control displays only when change the domain status for all domains on all machines using the **All Domains Inactive** box as described in the previous section.

To change Dispatcher Status for all processes on all machines from cleanup to inactive

1. Click the **Force Reset** button.
2. Click **Update**.

Setting Domain Grace Periods

This section describes how to set domain grace periods.

To set domain grace periods

1. Choose PeopleTools, Integration Broker, Monitor, Monitor Message.
2. Click the **Domain Status** page.
3. Enter a the domain grace period:
   a. To set one grace period to apply to domains on all machines:

      In the **Domain Criteria section**, in the **Grace Period for all Domains** field, enter a numeric value for the number of minutes for the grace period. Click **Update**.

   b. To set grace periods for individual domains:

      In the **Domain Status section**, for each domain enter a numeric value for the number of minutes for the grace period. Click **Update**.

Running Message Queries

When you have numerous messages, nodes, and channels in your messaging system keeping track of all of the definitions and routings can become an unwieldy task. The Queries page is designed so that the system administrator is able to gather information about the messaging system meta-data without having to access the message definitions, channel definitions, or node definitions in Application Designer.

The queries take advantage of the URL interface for PeopleSoft Query and Process Scheduler Report Manager.

Because PeopleSoft Query is involved, anyone running the queries requires the appropriate security permissions to do so.

These queries are not intended to be run every day or even on a regular basis. You might find that they are most useful for providing auditing information shortly after an implementation or upgrade. PeopleSoft Query enables you to view the information with Excel, which makes sharing the information with colleagues easier.

The following table contains descriptions of each of the available queries.

Query	*Query Name*	*Description*
Request Message Transactions	PT TRANS FOR MSG RQST	Shows the transactions that exist for a specific request message.
Response Message Transactions	PT TRANS FOR MSG RESP	Shows the transactions that exist for a specific response message.
Message Status at Node	PT MSGSTATUS AT NODE	Shows what messages are active or inactive in the local node.

Query	*Query Name*	*Description*
Messages that belong to a Channel	PT MSGS IN A CHANNEL	Shows what messages belong to each channel definition. For example, if you were curious as to what messages are grouped together for ordering and routing, run this query.
Channels and Messages at a Node	PT CHNLS MSGS AT A NODE	Reveals the message status for a node as well as showing which messages are associated with which channels to provide an overview of the messaging topology in a particular node/database.
Channels to Archive/Purge Messages	PT CHNL ARCH PURGE AT A NODE	Shows which channels should have their messages archived and which channels should have their messages purged. You can customize the query to prompt for criteria.

See Also

PeopleTools 8.4 PeopleBook: PeopleSoft Query

PeopleTools 8.4 PeopleBook: PeopleSoft Process Scheduler

Monitoring Asynchronous Message Details

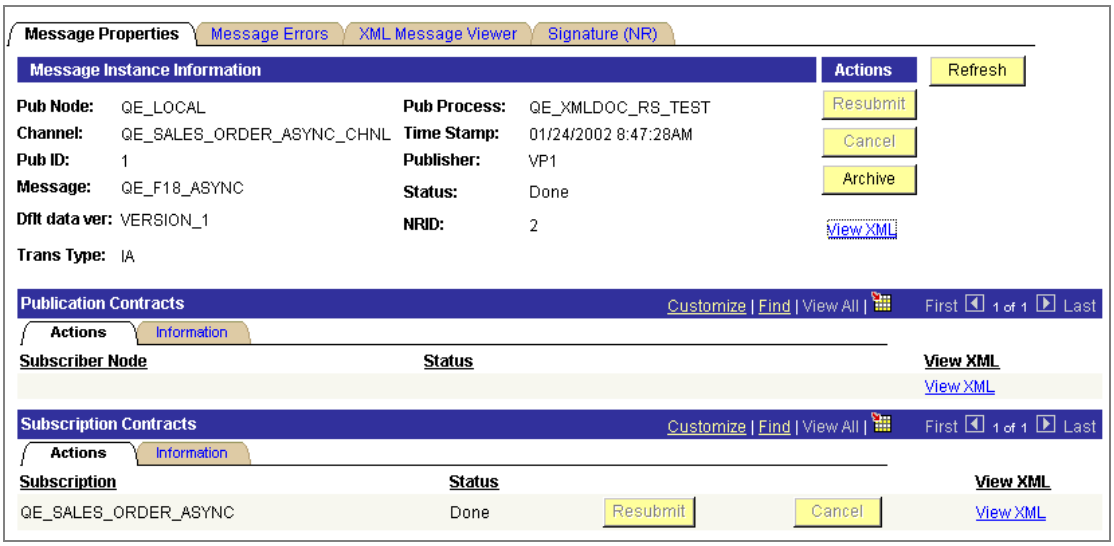
The Message Details component allows you to gather in-depth information about a specific asynchronous message. It also enables you to perform tasks such as correct errors and resubmit messages.

This section discusses how to:

- Open the Message Details component.
- Monitor asynchronous message details.
- View asynchronous message errors.
- View asynchronous messages in XML format.
- View nonrepudiation information for asynchronous messages.

Opening the Message Details Component

To open the Message Details component, use the PeopleTools menu structure and select **PeopleTools, Integration Broker, Monitor, Message Details**. The Message Detail component appears. By default, the Message Properties page is active.



PeopleTools, Integration Broker Monitor, Monitor, Message Details

The following sections describe how to use each page in the Message Details component.

Viewing Asynchronous Message Properties

The Message Properties tab displays information pertaining to the Message Instance and all associated Publication and Subscription Contracts.

There are three sections that appear on the Message Properties page. This section discusses the three grids and the information that appears in them. This section discusses:

- Message Instance Information section.
- Publication Contracts section.
- Subscription Contracts section.

Message Instance Information Section

The Messaging Instance Information section provides general information pertaining to a particular message to assist in troubleshooting.

Pub Node Identifies the node that published the message.

Channel Identifies the channel to which the message is associated.

Pub ID Identifies the Publication ID. Unique identifier for message. Applies to asynchronous messages only.

Message Identifies the message name.

Dflt data ver Identifies the default data version.

Trans Type Identifies the transaction type. Values are:

- *OA:* Outbound asynchronous.
- *IA*: Inbound asynchronous.

Pub Process Identifies the process that published the message; the name of the component that published the message.

Time Stamp Identifies the date and time that the message instance was last processed.

Publisher Identifies the publisher of the message. The publisher is usually the UserID of the person in the publishing system who triggered the message publication.

Status Identifies the status of the message, such as *Done*, *Error*, *Started*, and so on.

NRID Identifies a nonrepudiation ID. This ID is a unique number used to associate a message instance with the non-repudiation log.

The **Actions** column in the Message Instance section contains controls that apply to the Message Instance and all associated publication and subscription contracts.

Click the **Resubmit** button to resubmit a message for processing. This button is enabled when a message has a status of *Time Out*, *Error*, *Edited*, or *Cancelled*. If a message contains an error or has timed out, typically you can just correct the problem and resubmit the message. After you edit a message, the status becomes *Edited.* When you resubmit the message, the status changes, yet again, to *New*.

Click the **Cancel** button to cancel processing attempts for a message. This button is enabled with a message has a status of *New*, *Retry*, *Time Out*, *Error*, or *Edited.*

Click the **Archive** button to archive a message. This button is enabled when a message has a status of *Done* or *Cancelled.*

View XML Click the **View XML** link to view the XML data that was received for the selected message instance.

Publication Contracts Section

The **Actions** tab reveals all the nodes subscribing to a particular message and the current status of the publication contract, as in whether the publication has been successfully posted to the subscribing node.

If you click the **View XML** link, a new page opens and the XML for the selected message displays. If any transformations were applied for the Publication Contract, the transformed XML displays. Use the View XML link in the Message Instance section to see the original XML that was received.

You can edit the XML if you have the appropriate permissions, and if the message has the status of *New*, *Error*, *Retry*, *Timeout*, *Edited* or *Cancelled*. See Message Status Information

The **Information** tab contains the following information about the Publication Contract:

Subscriber Node	Identifies the name of the subscribing node.
Status	Identifies the processing status.
Time Stamp	Identifies the time that the system last modified the publication contract.
Retry Count	If the first attempt to deliver the message failed, this value reflects the number of times the system has attempted to redeliver the message.
Machine Name	Identifies the name of the local application server machine that processed the publication contract.
PID	Identifies the process ID on the local application server. It shows the PID of the PSMSGHND (handler) that created the contract.
NRID	Identifies the nonrepudiation ID (NRID). The NRID is a unique number used to associate a message instance with the nonrepudiation log.
Trans Type	Identifies the transaction type. Values are: • *OA:* Outbound asynchronous. • *IA*: Inbound asynchronous.
Request Message Name	Identifies the request message name. This message name can be different than the name in the message instance due to a relationship.
App. Msg Ver	Identifies the message version.

Subscription Contracts Section

The **Actions** tab reveals the status of a particular subscription contract.

If you click the **View XML** link, a new page opens and the XML for the selected message displays. If any transformations were applied for the Subscription Contract, the transformed XML displays. Use the View XML link in the Message Instance section to see the original XML that was received.

You can edit the XML if you have the appropriate permissions, and if the message has the status of *New*, *Error*, *Retry*, *Timeout*, *Edited* or *Cancelled*. See Message Status Information

The **Information** tab provides the following information.

Subscription	Identifies the subscription process name.
Status	Identifies the processing status.

Time Stamp	Identifies the time that the system last modified the subscription contract.
Retry Count	If the first attempt to subscribe to the message failed, this value reflects the number of times the system has attempted to subscribe to the message.
PID	Identifies the process ID on the server.
SubProcID	Identifies the subscription process ID associated with the subscription contract for identification purposes.
Trans Type	Identifies the transaction type. Values are: • *OA:* Outbound asynchronous. • *IA*: Inbound asynchronous.
App. Msg Ver	Identifies the message version.

Viewing Asynchronous Message Errors

The Message Errors page enables you to view any possible errors with message instances, publication contracts, and subscription contracts.

The tabs on this page let you see when an error occurred, what the error message is, and where the error occurred.

In addition, each section on the page provides the following control.

Click the **Download** button to download output to a spreadsheet.

Viewing Asynchronous Messages in XML Format

The XML Message Viewer page in the Message Details component enables you to view the raw XML code. This tab is where you can locate specific sections of the XML code.

Use the **Display Formatted** check box to adjust how the XML appears. In most cases, displaying the message in its formatted form is easier to read and to locate specific values or tags.

The XML Message Viewer page also features several links.

Edit XML	Use the **Edit XML** link to edit the XML displayed. The button is enabled provided you have the appropriate permissions and the status of the message is *New*, *Error*, *Retry*, *Timeout*, *Edited* or *Cancelled.*
Print XML	Use the **Print XML** link to print the XML. When you click the button, the XML appears in a new window. Use the Print command on your browser to print the page.

Viewing Nonrepudiation Information for Asynchronous Messages

The Signature (NR) page of the Message Details component displays nonreputiation information for asynchronous messages.

This page displays only if a message is sent with a signature, and displays the message signature in XML format.

Click the **Confirm** button to confirm the non-repudiation status.

Viewing Synchronous Message Details

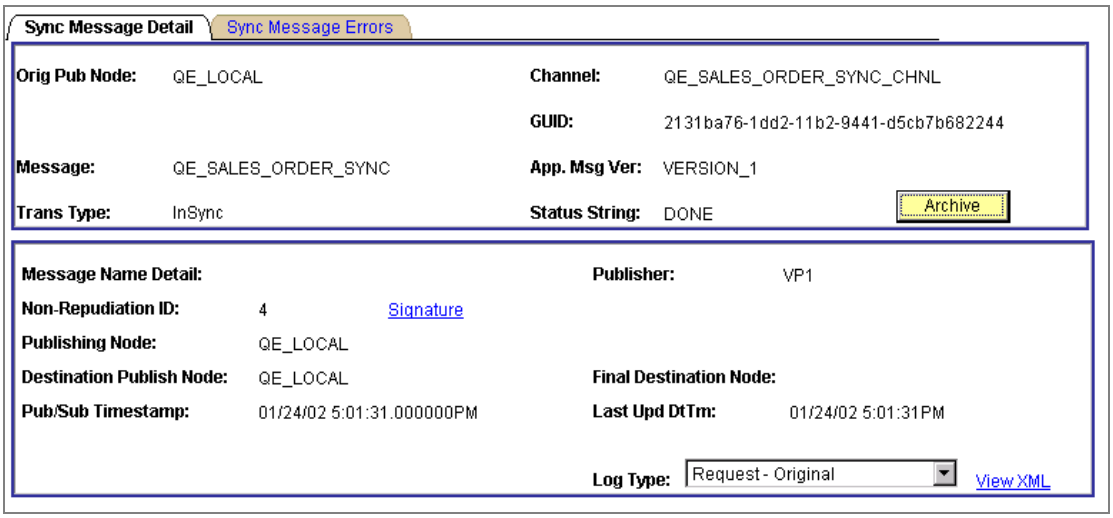
The Synchronous Details component allows you to gather in-depth information about a specific synchronous message. It also enables you to perform tasks such as view logs and correct errors.

This section discusses how to:

- Open the Synchronous Details component.
- View details about synchronous messages.
- View synchronous message errors.

Opening the Synchronous Details Component

To open the Synchronous Details component, use the PeopleTools menu structure and select **PeopleTools, Integration Broker, Monitor, Synchronous Details**. By default, the Sync Message Detail page is active.



PeopleTools, Integration Broker Monitor, Monitor, Synchronous Details

The following sections provide information about each page in the Synchronous Details component.

Viewing Synchronous Message Details

The Sync Message Detail Page provides read-only information about synchronous messages in your system. It also enables you to view signature information for a message if it was processed with non-repudiation logic.

In addition, you can use the Archive button to archive messages.

The section discusses:

- Message attributes.
- Viewing message signature information.

Message Attributes

You can view the following message attribute information on the Sync Message Detail page.

Note. For synchronous messages, you can view the full message detail in XML only if you have checked the Log Message Detail option for the transactions. To select this option, choose Node Definitions, Transactions, Edit (a transaction), Transaction Detail, Message.

Orig Pub Node	Identifies the node that published the message.
Channel	Identifies the channel to which the message is associated.
GUID	Identifies the Global Unique Identifier (GUID) that uniquely identifies each message. GUID applies to only to synchronous messages.
Message	Identifies the message name.
Version	Identifies the message version.
Trans Type	Identifies the transaction type. Values are: • *OutSync*: Outbound Synchronous • *InSync*: Inbound Synchronous
Status	Identifies the status of the message, such as Done, Error, Started, and so on.
Message Name Detail	Reserved for future use.
Publisher	Publisher of the message. This is usually the UserID of the person in the publishing system who triggered the message publication.

Time Stamp	Identifies the date and time that the message instance was last processed.
Publisher	Identifies the publisher of the message. This is usually the UserID of the person in the publishing system who triggered the message publication.
Non-Repudiation ID	Identifies a unique number used to associate a message instance with the non-repudiation log.
Publishing Node	Identifies the node that published the message.
Destination Publish Node	Identifies the name of the node where the message will be sent.
Pub/Sub Timestamp	Indicates the time and date of when the message was published.
Last Upd DtTm	Identifies the date and time the message was last updated.
Log Type	Select a value from the dropdown list and click the View XML link to view the corresponding information. Values are: • *Request Original*: Displays the original request data in XML format. • *Request Transformed*: Displays transformed request data, if applicable, in XML format. • *Response Original*: Displays the original response data in XML format. • *Response Transformed*: Displays the transformed response data, if applicable, in XML format.

Viewing Message Signature Information

If a message is sent with a Signature, a Signature link displays next to the Non-Repudiation ID number. When you click the Signature link, the message signature displays in XML. You can click the **Confirm** button to confirm the non-repudiation status. Click the Return button to return to the Sync Message Details page.

Viewing Synchronous Message Errors

The Sync Error page in the Synchronous Details component provides read-only information about any error related to synchronous messages.

The information that displays on the page lets you know the original publishing node of the message, the channel, the message name, the transaction type, and more.

The tabs on this page let you view error message details such as what type of error occurred, and when and where the error occurred.

Running Batch Error Notification Processes

This section describes batch processes that PeopleSoft provides to assist in notifying you of errors. To access the interface to invoke each of these programs, select **PeopleTools, Integration Broker, Monitor, Error Notification**.

Although you can easily use the Integration Broker Monitor to scan your system for messages, that approach requires you to launch Integration Broker Monitor on a scheduled basis to search for any issues affecting your messaging system. Rather than using this labor-intensive approach, PeopleSoft provides an Application Engine batch program, named PT_AMM_WF, that you schedule to run on a recurring basis.

Note. You can use PT_AMM_WF to notify you of errors relating to asynchronous messages only.

To enable the workflow notification functionality, you need to have the following items in place within security definitions:

- Grant access to the PT_AMM_DUMMY component interface using PeopleTools, Maintain Security, Use, Permission Lists, Component Interface.
- Assign Users to the APP_MSG_ADMINISTRATOR role using PeopleTools, Maintain Security, Use, Roles. PeopleSoft delivers this role "out of the box."
- Users with assigned to the APP_MSG_ADMINISTRATOR role need to have their e-mail address added to their user profile so that they can receive the notification. You complete this task using PeopleTools, Maintain Security, Use, User Profiles.

The following table provides a step-by-step process that describes the information the Application Engine program, PT_AMM_WF, scans for, how it notifies administrators, and what administrators should do after receiving an error notification.

Step	*Task*	*Description*
1	Query Message Queues	The Application Engine program, PT_AMM_WF, scans the following messaging queues in the database in search of messages with a status of either "ERROR" or "TIMEOUT". Publications Queue Publications Contracts Queue Subscriptions Contracts Queue
2	Trigger Workflow	Upon encountering a message status of either "ERROR" or "TIMEOUT", PT_AMM_WF triggers a Workflow notification, which the system sends, by default, to all users that qualify for the query Role APP_MSG_ADMINISTRATOR at runtime. PeopleSoft supplies this Role as part of the delivered system data. As delivered, the query for this role associates a message with a user through the message's Channel Name property. All users that have at least read-access to the message channel get notified.

Step	Task	Description
3	Resolve Issue	After an administrator receives the workflow notification by email, they check their worklist to find a new worklist item reflecting the problematic message. To access the message, the administrator clicks the item in the worklist. The link leads to the Message Details component. The component is presented with the errored message loaded.

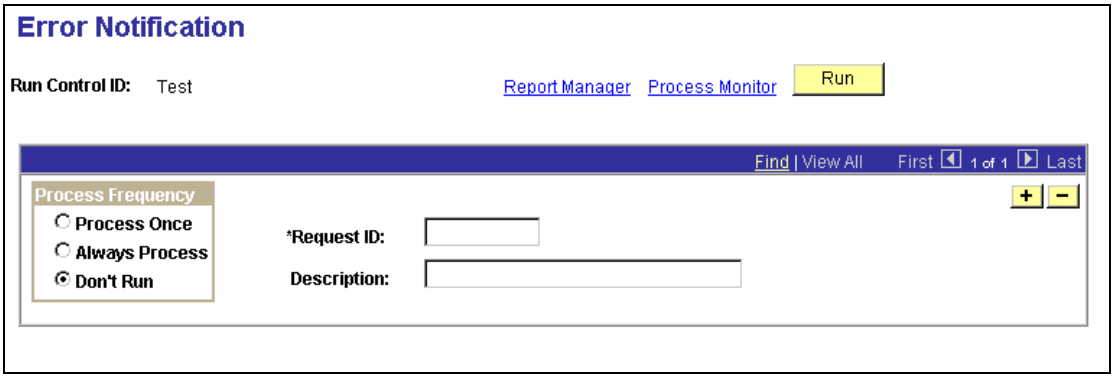
Running the Batch Program PT_AMM_WF

To run PT_AMM_WF

1. Select **PeopleTools, Application Integration Broker Monitor, Process, Error Notification.**

2. Select an existing **Run Control ID**, or add a new one using the **Add** button.

   The Error Notification page appears.



   Error Notification

3. Select a **Process Frequency**.

   You have the following choices:

   - **Process Once.** If you only want run PT_AMM_WF once, manually, on and as needed basis, select this option.
   - **Process Always**. If you want PT_AMM_WF to always run, constantly, select this option.
   - **Don't Run**. If you want to disable a recurring PT_AMM_WF run, select this option.

4. Add a **Request ID** and **Description**.

   This is where you add attributes to uniquely identify a Run Control. You only see it when you have a list of Run Controls.

5. Click **Run.**

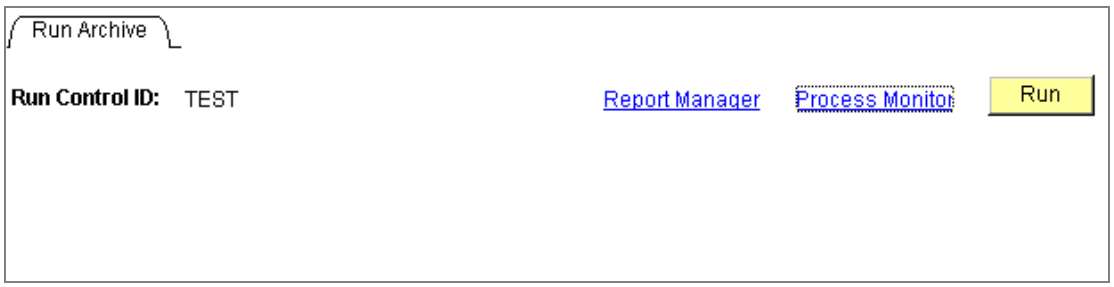
6. Click **OK** on the Process Scheduler Request page to submit the process.

Running Batch Message Archiving Processes

For performance and general maintenance reasons you may want to archive older messages to clear space on your live messaging tables.

PeopleSoft provides an Application Engine program, APPMSGARCH, which scans all of the messaging tables in the system and removes all messages with a status of *Done*. You use the Run Archive page to invoke the archive process.

Note. APPMSGARCH archives asynchronous messages only.



PeopleTools, Integration Broker, Monitor, Archive Messages.

Note. Using APPMSGARCH to archive message data is the batch approach. You can also archive individual messages online using the Integration Broker Monitor.

Running the Batch Program APPMSGARCH

To run APPMSGARCH

1. Select **PeopleTools, Integration Broker, Monitor, Archive Messages.**
2. Select an existing Run Control ID or add a new one.

   The Run Archive page appears.
3. Make sure the appropriate Run Control ID appears on the page, and click **Run.**
4. On the Process Scheduler Request page make the appropriate selections, and click **OK.**

See Also

PeopleTools 8.4 PeopleBook: PeopleSoft Process Scheduler, "Using Process Monitor"

PeopleTools 8.4 PeopleBook: PeopleSoft Process Scheduler, "Using Report Manager"

Purging Messaging Tables

PeopleSoft provides a collection of Data Mover scripts that you can run to purge the messaging tables within a database. These scripts reside in the PS_HOME\scripts directory on your file server. The following table describes what each script is designed to accomplish.

AppMsgPurgeAll.dms	Deletes the queue data from every archive or live message table in the database. Typically, you run this script after an upgrade or while switching from your demonstration to your production environment.
AppMsgPurgeArchive.dms	Deletes the queue data from every archive message table in the database.
AppMsgPurgeLive.dms	Deletes the queue data from every live message table in the database.

Using the Message Monitor Component Interface

Not all of the Integration Broker Monitor functionality is exposed. PeopleSoft provides a collection of inquiry methods you can access with a component interface.

The name of the component interface you use to extract information from the Integration Broker Monitor is MSGSTATUSSUMMARY. The output of the component interface reveals the amount of contracts that are in the queue. The contracts appear grouped by status and message or grouped by status and channel.

You can use the following user-defined methods to extract information:

- FillPubConByMsg()
- FillPubConByChannel()
- FillSubConByMsg()
- FillSubConByChannel()

Following example shows some sample of ASP code that accesses the MSGSTATUSSUMMARY component interface with COM.

```
'Create a peoplesoft session
Set oSession = server.CreateObject ("PeopleSoft.Session")
nStatus = oSession.Connect(1, oConnectString, oUserName, oPassword,0)

'Get the skeleton of the APPMSGMON CI
Set oCI = oSession.GetCompIntfc("MSGSTATUSSUMMARY")

'get an instance of the CI
nStatus = oCI.Get()
'execute the method to fill the collection
```

```
If oChoice = 1 then
   nStatus = oCI.FillPubConByChannel()
   'Set oRows to the properties collection
   Set oRows = oCI.PubConByChannel
End If

If oChoice = 2 then
   nStatus = oCI.FillPubConByMsg()
   'Set oRows to the properties collection
   Set oRows = oCI.PubConBymsg
End If

If oChoice = 3 then
   nStatus = oCI.FillSubConByChannel()
   'Set oRows to the properties collection
   Set oRows = oCI.SubConByChannel
End If

If oChoice = 4 then
   nStatus = oCI.FillSubConByMsg()
   'Set oRows to the properties collection
   Set oRows = oCI.SubConByMsg
End If
```

See Also

PeopleTools 8.4 PeopleBook: PeopleSoft Component Interfaces

CHAPTER 9

Understanding Error Handling, Logging, Tracing and Debugging

Error handing, logging, tracing and debugging in conjunction with the Integration Broker can occur on the Integration Gateway, Application Server or Application Engine—depending on the type and location of processing.

This chapter discusses:

- Integration Gateway error handling.
- Integration Gateway message and error logging.
- Application Server logging and tracing.
- Application Engine tracing.
- Debugging subscription PeopleCode.
- Integration Broker debugging quick reference information.

Integration Gateway Error Handling

Error handling is an Integration Gateway service that assists connectors to manage errors that occur during processing. Errors on the Integration Gateway are handled by both target connectors and listening connectors.

This section discusses:

- Target Connector error handling.
- Listening Connector error handling.

Target Connector Error Handling

The Target Connector Interface specifies the methods that target connectors must implement in order for the Integration Gateway to manage them. Tied into these methods is a set of standard exceptions that target connectors throw if and when they experience errors during processing.

Listening connectors or the Gateway Manager catch these exceptions and provide an appropriate implementation for each. The Gateway Manager catches the exceptions when the source of the message is an Integration Engine using the PeopleSoft Listening Connector. Otherwise, listening connectors are responsible for handling exceptions thrown during processing.

Listening Connector Error Handling

Unlike target connectors, listening connectors are not managed by the Gateway Manager and therefore do not adhere to any interface. However, a listening connector must invoke the Gateway Manager to pass a message from the Integration Gateway to the Integration Engine. The Gateway Manager has a number of predefined exceptions it can throw, and throws them where appropriate in the Integration Broker implementation.

In general, exceptions are thrown in a target connector and caught by a listening connector. As a result, a listening connector must catch these exceptions and handle them as appropriate, typically by generating an error message and sending it back to the requestor.

Integration Gateway Exception Types

Integration Gateway errors fall within two categories. This section provides information about these categories and their exceptions, and discusses:

- Standard exceptions
- Java exceptions

Standard Exceptions

The following table lists and describes standard error and exception types that the Integration Gateway, target connectors and listening connectors can handle.

DuplicateMessageException	Thrown when a target connector is aware that it is processing a message that has already been processed. This is usually discovered based on an error attained from the external system being contacted. Of all the connectors delivered by PeopleSoft, only PeopleSoft 8.1 Target Connector (PSFT81TARGET) can throw this exception. Target connectors are not required to have the ability to throw this exception.
ExternalApplicationException	The message reached its intended destination but could not be processed.

	Determining that the destination could not process a certain message requires significant knowledge of the destination system, which a target connector may or may not have. Whenever possible, a target connector should make an attempt to determine this situation, otherwise this task will have to be decentralized and handled outside of the Integration Gateway. For example, the HTTP Target Connector (HTTPTARGET) throws this exception when the external system returns an HTTP system code of *500*.
ExternalSystemContactException	The target connector cannot properly establish a connection with the intended destination. One of the most common exceptions. When thrown during an asynchronous transaction, the Integration Broker will try to resend the message until successful.
GeneralFrameworkException	General error in the Integration Gateway.
InvalidMessageException	Thrown when a connector or the Gateway Manager determines that the message cannot be processed because of missing or erroneous information in a request or response. One of the most common exceptions.
MessageMarshallingException	Gateway Services' attempt to get information from an IBRequest or IBResponse failed. Can occur when the Gateway Services attempts to access a content section of a document using an out-of-range index from one of the following methods: • GetContentSectionAt(index) • GetContentSectionInfoAt(index) • RemoveContentSectionAt(index) If you try to access IBRequest or IBResponse with an out-of-range index using any of these methods, this exception is thrown automatically and processing is interrupted.
MessageUnmarshallingException	Gateway Services' attempt to build an IBRequest or IBResponse failed. Failure can occur when: • Instantiating an IBRequest/IBResponse from a MIME where the message sent does not comply to the PeopleSoft Mime format.

- Instantiating an IBRequest using the PS_XML format passing an invalid PS_XML message. Typically from the HTTP Listening Connector.
- Setting invalid values to methods such as setTransactionID or setMessageType.

These failures cause the Integration Gateway to throw this exception automatically, and processing is interrupted.

Java Exceptions

In addition, Integration Gateway target connectors and listening connectors can handle miscellaneous Java exceptions, such as NullPointerException, ArrayOutOfBoundsException, and so forth.

Integration Gateway Message and Error Logging

Error and message logging is a Gateway Service you can use to monitor messages that flow through the Integration Gateway.

Logging takes place within both target and listening connectors. Connectors have the ability to log all message requests and responses. As a result, you can use logging to:

- Track message flow.
- Troubleshoot processing errors.

By default the Integration Gateway is configured to log everything, including all errors, warnings, important, standard and low importance information.

You set up message and error logging using the IntegrationGateway.properties file. This configuration file features a Logging Setting section where you can view or change the default settings items such as the level of gateway logging, where the system writes log files, the maximum size of the log file, the number of file backups or archives to keep, and so forth.

This section discusses:

- Integration Gateway Message logging.
- Integration Gateway Error logging.
- Sample logging from a listening connector.
- Sample logging from a target connector.

See Also

“Using the Integration Gateway,” Setting Integration Gateway Properties

Integration Gateway Message Logging

This section provides an overview of Integration Gateway message logging and discusses:

- The default location of the Integration Gateway message log.
- Information recorded in the Integration Gateway message log.
- Message logging in target connectors.
- Message logging in listening connectors.
- The method (**logMessage**) used to invoke Integration Gateway message logging, and its parameters.

Integration Gateway Message Log Location

The default location of the Integration Gateway Message Log depends on the Web server you are using:

- WebLogic:

  c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\msgLog.html

- WebSphere:

  c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\msgLog.html

You can change the location of the log in the IntegrationGateway.properties file.

Information Recorded in the Integration Gateway Message Log

Message logging records the following information for messages that pass through the Integration Gateway:

- Time and date.
- Message description.
- Content of the passed message object.
- Message level.

Message Logging in Target Connectors

Message logging in a target connector occurs:

- Prior to delivering the request to the external system. The connector logs the request in the format in which the external system delivered it.

  For example, an HTTP Target Connector logs the exact HTTP output stream request. The PeopleSoft Target Connector logs the MIME request to be sent to the Integration Engine.

- After it receives a response from the external system. The connector logs the response in the format in which it is received.

For example, an HTTP Target Connector logs the exact HTTP input stream response. The PeopleSoft Target Connector logs the MIME response received from the Integration Engine.

Message Logging in Listening Connectors

Message logging in a listening connector occurs:

- At the point the request gets into the system. The connector logs the request in the format in which the sending system delivers it.

As an example, the HTTP Listening Connector logs the exact HTTP input stream request. The PeopleSoft Listening Connector logs the MIME request received from the Integration Engine.

- Following the delivery of a response to the requestor system. The connector logs the response in the format in which it was delivered.

For example, the HTTP Listening Connector logs the exact HTTP output stream response. The PeopleSoft Listening Connector logs the MIME response sent back to the Integration Engine.

Integration Gateway Message Logging Methods and Parameters

The method invoked for Integration Gateway message logging is **logMessage**:

```
logMessage(String Description, Object message, int MessageLevel).
```

The logMessage method has the following parameters.

Description	Specify a description as a string.
Object	Specify the message object. Typically this object will be an IBRequest or IBResponse. If another object is passed, the **toString** method is invoked for the object, and the result is logged.
MessageLevel	Specify whether the log gets written to permanent storage. This parameter takes an integer value. Sets the relative importance of the information you are logging. The ig.log.level property setting in the IntegrationGateway.properties file determines the log level currently in effect. If the MessageLevel value passed in the property is less than or equal to the ig.log.level property setting, the message is written to the log file.

Values are:

3: Important Information
4: Standard Information
5: Low Importance Information

The ig.messageLog.filename property in the IntegrationGateway.properties file determines the log file location.

Integration Gateway Error Logging

This section provides an overview of Integration Gateway error logging and discusses:

- The default location of the Integration Gateway error log.
- Information recorded in the Integration Gateway error log.
- The method (**logError**) used to invoke Integration Gateway error logging, and its parameters.

Integration Gateway Error Log Location

The default location of the Integration Gateway Error Log depends on the Web server you are using:

- WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\errorLog.html

- WebSphere:

c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\errorLog.html

You can change the location of the log in the IntegrationGateway.properties file.

Information Recorded in the Integration Gateway Error Log

Error logging captures processing errors that occur in the Integration Gateway. When an error occurs, the following information is logged.

- Error level.
- Description.
- Message Catalog entry Information.
- Stack trace identifying the problem.
- IBRequest and IBResponse. (If available.)

Integration Gateway Error Logging Methods and Parameters

The method invoked for Integration Gateway error logging is **logError**:

```
logError (String Description, IBRequest, IBResponse, int ErrorLevel,
Throwable)
```

The **logError** method features the following parameters.

Description	Specify a description as a string.
IBRequest	Specify the IBRequest for this transaction, if available. If not available, simply pass *Null*.
IBResponse	Specify the IBResponse for this transaction, if available. If not available, simply pass *Null*.
ErrorLevel	Specify whether the log gets written to permanent storage. Determines the severity of the error. The ig.log.level property in the IntegrationGateway.properties file determines the log level currently in effect. If the ErrorLevel value passed in here is less than or equal to the ig.log.level property setting, the error is written to the log file. Values are: • *-100*: Language exception • *1*: Standard gateway exception • *2*: Warning The ig.errorLog.filename property in the IntegrationGateway.properties file determines the log file location.
Throwable	Specify the Java exception or error associated with the error. This is used to log the stackTrace associated with the error.

The Gateway Manager and delivered listening connectors feature built-in error logging which invokes the logError method. The delivered target connectors do not feature built-in error logging, and instead throw any errors that occur during processing to the Gateway Manager or listening connectors, where they are handled or logged.

Application Server Logging and Tracing

The PeopleSoft Application Server Administration menu enables you to:

- View application server and Tuxedo log files. See *PeopleTools 8.4 PeopleSoft Architecture PeopleBook,* “Understanding PSADMIN Menus” Edit configuration/log files Menu

- Trace SQL and PeopleCode on your domain(s). See *PeopleTools 8.4 PeopleSoft Architecture PeopleBook,* "Understanding Application Server Domain Parameters," Trace
- Set the level of network tracing (log fence). See *PeopleTools 8.4 PeopleSoft Architecture PeopleBook,* "Understanding Application Server Domain Parameters,"Domain Settings
- View the certificate authentication logs, including information about mismatched Distinguished Names and certificates not in the database. This information is contained in the APPSRV.LOG.

Application Engine Tracing

The PeopleSoft Application Engine tracing functionality enables you to monitor the performance of transforms on your implementation of the Integration Broker.

See Also

"Applying Transformation, Translation and Filtering"

PeopleTools 8.4 Application Engine PeopleBook, "Tracing Application Engine Programs"

Debugging Subscription PeopleCode

You can debug subscription PeopleCode using the following instructions.

To debug Subscription PeopleCode:

1. Start Application Designer on the server machine that is processing the subscription.

   You must use the same User ID and database as that machine has had its Application Servers configured to.
2. Start debug mode.

   When the subscription server executes the subscription PeopleCode, the debugger will start.

Integration Broker Debugging Quick Reference

Refer to the following table for a quick reference for Integration Broker debugging.

Area/ Suspected Problem	*Debugging Suggestion*
Application server exceptions.	Check the application server log: <install directory>\appserv\<Domain>\LOGS\ appsrv.log

Area/ Suspected Problem	*Debugging Suggestion*
Message handlers not up and running.	Check the application server domain status or queue status in psadmin: Go to **Domain Status**, **Server Status or Domain Status**, **Queue Status**.
Integration Gateway.	Check the IntegrationGateway.properties file and verify property settings. The default location of this file depends on the Web server: WebLogic: • c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Web-inf\integrationGateway.properties WebSphere • c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Web-inf\integrationGateway.properties
	Check the Integration Gateway Error Log. The default location of this file depends on the Web server: WebLogic: • c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\errorLog.html WebSphere: • c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\errorLog.html
	Check the Integration Gateway Message Log. The default location of this file depends on the Web server: WebLogic: • c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\msgLog.html WebSphere: • c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\msgLog.html
Message channels paused.	Check Integration Broker Monitor. Select **PeopleTools**, **Integration Broker**, **Monitor**, **Monitor Message, Channel Status**.
Node paused.	Check Integration Broker Monitor. Select **PeopleTools**, **Integration Broker**, **Monitor**, **Monitor Message, Node Status**.
Incorrect Gateway URL.	Integration Broker Monitor. Select **PeopleTools**, **Integration Broker**, **Monitor**, **Monitor Message, Node Status.**
Node inactive.	Check the node definition. Select **PeopleTools**, **Integration Broker**, **Node Definitions.**

Area/ Suspected Problem	*Debugging Suggestion*
Subscription PeopleCode missing or incorrect.	Check Integration Broker Monitor. Select **PeopleTools**, **Integration Broker**, **Monitor**, **Monitor Message, Sub Contracts**. Check the Queue Information tab. Also check **Application Designer**, **Message Designer**.
Message inactive.	Check **Application Designer**, **Message Designer**.
Transform problems.	Check Application Designer, Application Engine.
	For before and after images, check the Message Monitor. • For asynchronous messages, select **Integration Broker**, **Monitor**, **Message Details**. On the **Message Properties** tab, click **View XML** for the Publication Contract or Subscription Contract. • For synchronous messages, select **Integration Broker**, **Monitor**, **Synchronous Details**. On the **Sync Message Detail** tab, use the **Log Type** dropdown list to select **Request Transformed** or **Response Transformed** and then click **View XML**.
	Check that theTraceAE flag in the following directory is equal to *8192*: <install directory>\appserv\<Domain>\psappsrv.cfg Setting the TraceAE flag in the psappsrv.cfg file instructs the application server to generate a transformation trace log which will show: • The original XML structure as it entered the transformation engine. • The output of the XML as it passed through each step of the Transform program. The trace file with the extension AET is written to the following directory: <install directory>\appserv\<Domain>\LOGS\ <operID>_<machine name>.AET

CHAPTER 10

Applying Transformation, Translation and Filtering

This chapter explains how to:

- Understand the PeopleSoft base message format.
- Develop Application Engine transform programs.
- Filter messages based on content.
- Apply transformations to modify message structure.
- Perform data translation on messages.

Understanding Transformation, Translation and Filtering

Transformation, translation and filtering are all accomplished by applying an Application Engine *transform program* to an inbound or outbound message. You can use transform programs to do any of the following:

- Apply a *transformation* to a message to make its structure comply with the target system's requirements.
- Perform a *data translation* on a message so its data is represented according to the target system's conventions.
  - Simple translation might be required if the two systems use different values to represent the same information for a given field.
  - Complex translation might involve augmenting or replacing groups of fields with a completely different structure and encoding.
- Determine whether to pass a message through to its target, by *filtering* it based on its content.

If your PeopleSoft application uses the PeopleCode XmlDoc or SoapDoc classes to generate or process a message, the message probably doesn't adhere to the PeopleSoft base message format. You must make sure all participating nodes agree on a format, or employ transformations to accommodate the variations from node to node.

Note. To develop a transform program, you must know in detail the initial structure and possibly the content of the message you're working with, as well as the structure (and content) of the result you want to achieve. See Understanding the PeopleSoft Base Message Format for details about PeopleSoft's standard message structure.

Transformation, translation or filtering can be necessary for messages sent between two Integration Broker nodes, or between an Integration Broker node and a third party application. Any participating node with Integration Broker installed — the source, the target, or a hub — can apply a transform program to a given message.

You specify which transform program to apply as part of a *transaction modifier*, which is associated with a *relationship definition*. See "Administering Relationships" for more information about defining and using transaction modifiers.

Note. With Integration Broker, the term *node* is used to refer to a system or application participating in an integration, but in this chapter a *node* is also a structural element in an XML document. The context in which we use the term should make its meaning clear.

What Transform Programs Can't Modify

- Transmission protocols: To handle different protocols, configure the sending and receiving a target connector that supports your required protocol, or develop a new connector. See "Using the Integration Gateway" and "Using the Integration Broker Connector SDK."

- Character set encoding: This is handled by PeopleSoft's globalization system. See *PeopleTools PeopleBook: Globalization,* "Character Sets and Language Input/Output."

Third Party Considerations

Integration Broker requires the message data it handles to conform to a minimum XML message structure — the PeopleSoft base message format. When no transformation is applied, applications using Integration Broker send, and expect to receive, messages containing data in this format.

Third party applications have two choices when exchanging messages with PeopleSoft applications:

- Send and receive messages that comply with the base message format. See Understanding the PeopleSoft Base Message Format.

- Employ a transformation at the PeopleSoft end of each transaction to convert messages to or from the base message format. See Applying Transformations.

Understanding the PeopleSoft Base Message Format

This section describes the basic structure of the data section of a PeopleSoft message. Third party applications must comply with this base message format if *both* of the following are true:

- Your PeopleSoft application uses the PeopleCode Message and Rowset classes to generate and send, or receive and process messages with Integration Broker.
- You don't wish to employ Integration Broker transformations to accommodate the third party application.

Note. All PeopleSoft applications that use Integration Broker or application messaging automatically comply with the PeopleSoft base message format; no further compliance is required.

Timestamp Format

The PeopleSoft format for all timestamps is ISO-8601:

```
CCYY-MM-DDTHH:MM:SS.ssssss+/-hhmm
```

Note. ISO format specifies that the +/-hhmm parameter is optional, but PeopleSoft requires it. All date and time stamps in the header and in the body of the message must include the GMT offset as +/-hhmm. This ensures that the timestamp is correctly understood by the receiving application.

The FieldTypes Section

Each message sent by an Integration Broker system includes *fieldtype* information. Fieldtype information conveys the name of each message record and its constituent fields, along with each field's data type. Your receiving application can use this information to validate the data types. The fieldtype information is contained in the *FieldTypes* section of the message.

There are two FieldTypes tags:

- Each record tag consists of the name of a record, followed by a *class* attribute with a single valid value — ***"R"***. The record tag encloses that record's field tags.
- Each field tag consists of the name of a field, followed by a *type* attribute with three valid values: ***"CHAR"*** for a character field, ***"DATE"*** for a date field, and ***"NUMBER"*** for a numeric field.

A Simple FieldTypes Template

```
<FieldTypes>
```

```
   <recordname1 class="R">
      <fieldname1 type="CHAR"/>
      <fieldname2 type="DATE"/>
      <fieldname3 type="NUMBER"/>
   </recordname1>
   <recordname2 class="R">
      <fieldname4 type="NUMBER"/>
   </recordname2>
</FieldTypes>
```

Note. Third party sending applications must include a valid FieldTypes section in each message. The PeopleSoft system expects fieldtype information for each record and field in the message.

The MsgData Section

In addition to fieldtype information, each message sent by an Integration Broker system of course includes actual data, in the *MsgData* section of the message. The MsgData section is structured as follows:

- Between the MsgData tags is one or more *Transaction* sections. Each transaction represents one row of data.
- Between the Transaction tags is a rowset hierarchy of records and fields. Within the level 0 record tags are the fields for that record, followed by any level 1 records.
- The last record within a transaction is a fully specified *PSCAMA* record, which provides information about the entire transaction.
- For a record at each level, within its record tags are that record's fields, followed by any records at the next lower level.
- Immediately following the closing tag of *every* record below level 0 is a PSCAMA record containing only the AUDIT_ACTN field to specify the action for that record.

A Simple MsgData Template

Note. The PSCAMA PUBLISH_RULE_ID and MSGNODENAME fields (shown in bold) are used internally by certain PeopleSoft utilities, but third party systems can generally ignore them and need not include them in messages.

```
<MsgData>
   <Transaction>
      <level0recname1 class="R">
         <fieldname1>value</fieldname1>
         <fieldname2>value</fieldname2>
```

```
            <level1recname1 class="R">
               <fieldname3>value</fieldname3>
               <fieldname4>value</fieldname4>
            </level1recname1>
            <PSCAMA class="R">
               <AUDIT_ACTN>value</AUDIT_ACTN>
            </PSCAMA>
            <level1recname2 class="R">
               <fieldname5>value</fieldname5>
            </level1recname2>
            <PSCAMA class="R">
               <AUDIT_ACTN>value</AUDIT_ACTN>
            </PSCAMA>
         </level0recname1>
         <level0recname2 class="R">
            <fieldname6>value</fieldname6>
         </level0recname2>
         <PSCAMA class="R">
            <LANGUAGE_CD>value</LANGUAGE_CD>
            <AUDIT_ACTN>value</AUDIT_ACTN>
            <BASE_LANGUAGE_CD>value</BASE_LANGUAGE_CD>
            <MSG_SEQ_FLG>value</MSG_SEQ_FLG>
            <PROCESS_INSTANCE>value</PROCESS_INSTANCE>
            <PUBLISH_RULE_ID>value</PUBLISH_RULE_ID>
            <MSGNODENAME>value</MSGNODENAME>
         </PSCAMA>
      <Transaction>
</MsgData>
```

See Also

"Sending and Receiving Messages," Understanding PSCAMA

A Message Example

The message data is enclosed in a tag with the name of the message, and consists of exactly one FieldTypes section followed by one MsgData section. The FieldTypes section describes the records and fields that appear in the MsgData section, which contains the actual data.

Note. The PSCAMA record requires fieldtype information just like any other record.

```
<SDK_BUS_EXP_APPR_MSG>
    <FieldTypes>
        <SDK_BUS_EXP_PER class="R">
            <SDK_EMPLID type="CHAR"/>
```

```
                <SDK_EXP_PER_DT type="DATE"/>
                <SDK_SUBMIT_FLG type="CHAR"/>
                <SDK_INTL_FLG type="CHAR"/>
                <SDK_APPR_STATUS type="CHAR"/>
                <SDK_APPR_INSTANCE type="NUMBER"/>
                <SDK_DESCR type="CHAR"/>
                <SDK_COMMENTS type="CHAR"/>
            </SDK_BUS_EXP_PER>
            <SDK_DERIVED class="R">
                <SDK_BUS_EXP_SUM type="NUMBER"/>
            </SDK_DERIVED>
            <SDK_BUS_EXP_DTL class="R">
                <SDK_CHARGE_DT type="DATE"/>
                <SDK_EXPENSE_CD type="CHAR"/>
                <SDK_EXPENSE_AMT type="NUMBER"/>
                <SDK_CURRENCY_CD type="CHAR"/>
                <SDK_BUS_PURPOSE type="CHAR"/>
                <SDK_DEPTID type="CHAR"/>
            </SDK_BUS_EXP_DTL>
            <PSCAMA class="R">
                <LANGUAGE_CD type="CHAR"/>
                <AUDIT_ACTN type="CHAR"/>
                <BASE_LANGUAGE_CD type="CHAR"/>
                <MSG_SEQ_FLG type="CHAR"/>
                <PROCESS_INSTANCE type="NUMBER"/>
            </PSCAMA>
        </FieldTypes>
        <MsgData>
            <Transaction>
                <SDK_BUS_EXP_PER class="R">
                    <SDK_EMPLID>8001</SDK_EMPLID>
                    <SDK_EXP_PER_DT>1998-08-22</SDK_EXP_PER_DT>
                    <SDK_SUBMIT_FLG>N</SDK_SUBMIT_FLG>
                    <SDK_INTL_FLG>N</SDK_INTL_FLG>
                    <SDK_APPR_STATUS>P</SDK_APPR_STATUS>
                    <SDK_APPR_INSTANCE>0</SDK_APPR_INSTANCE>
                    <SDK_DESCR>Regional Users Group Meeting</SDK_DESCR>
                    <SDK_COMMENTS>Attending Northeast Regional Users Group Meeting
and presented new release functionality.</SDK_COMMENTS>
                    <SDK_BUS_EXP_DTL class="R">
                        <SDK_CHARGE_DT>1998-08-22</SDK_CHARGE_DT>
                        <SDK_EXPENSE_CD>10</SDK_EXPENSE_CD>
                        <SDK_EXPENSE_AMT>45.690</SDK_EXPENSE_AMT>
                        <SDK_CURRENCY_CD>USD</SDK_CURRENCY_CD>
                        <SDK_BUS_PURPOSE>Drive to Meeting</SDK_BUS_PURPOSE>
                        <SDK_DEPTID>10100</SDK_DEPTID>
```

```
                </SDK_BUS_EXP_DTL>
                <PSCAMA class="R">
                    <AUDIT_ACTN>A</AUDIT_ACTN>
                </PSCAMA>
                <SDK_BUS_EXP_DTL class="R">
                    <SDK_CHARGE_DT>1998-08-22</SDK_CHARGE_DT>
                    <SDK_EXPENSE_CD>09</SDK_EXPENSE_CD>
                    <SDK_EXPENSE_AMT>12.440</SDK_EXPENSE_AMT>
                    <SDK_CURRENCY_CD>USD</SDK_CURRENCY_CD>
                    <SDK_BUS_PURPOSE>City Parking</SDK_BUS_PURPOSE>
                    <SDK_DEPTID>10100</SDK_DEPTID>
                </SDK_BUS_EXP_DTL>
                <PSCAMA class="R">
                    <AUDIT_ACTN>A</AUDIT_ACTN>
                </PSCAMA>
            </SDK_BUS_EXP_PER>
            <SDK_DERIVED class="R">
                <SDK_BUS_EXP_SUM>58.13</SDK_BUS_EXP_SUM>
            </SDK_DERIVED>
            <PSCAMA class="R">
                <LANGUAGE_CD>ENG</LANGUAGE_CD>
                <AUDIT_ACTN>A</AUDIT_ACTN>
                <BASE_LANGUAGE_CD>ENG</BASE_LANGUAGE_CD>
                <MSG_SEQ_FLG></MSG_SEQ_FLG>
                <PROCESS_INSTANCE>0</PROCESS_INSTANCE>
            </PSCAMA>
        </Transaction>
    </MsgData>
</SDK_BUS_EXP_APPR_MSG>
```

Adding Nonrepudiation Signatures

If you're working with a nonrepudiated message, its signature must be located at the same level as the message data. The message doesn't need to be formatted with the PeopleSoft rowset hierarchy, as long as it's enclosed in valid XML and has the signature section as specified by the World Wide Web Consortium (W3C). The following template describes a nonrepudiation signature alongside the PeopleSoft base format message data it represents:

```
<psft_message_name>
   <FieldTypes>
      ...
   </FieldTypes>
   <MsgData>
      ...
   </MsgData>
</psft_message_name>
```

```
<Signature>
   <SignedInfo>
      (CanonicalizationMethod)
      (SignatureMethod)
      (<Reference (URI=)? >
         (Transforms)?
         (DigestMethod)
         (DigestValue)
      </Reference>)+
   </SignedInfo>
   (SignatureValue)
   (KeyInfo)?
   (Object)*
</Signature>
```

See "Defining Messages," Configuring Message Properties for more information about nonrepudiation. See http://www.w3.org/TR/xmldsig-core/ for details about the W3C's proposed standard for XML Signature Syntax and Processing.

Important! Integration Broker assumes all signatures use line feeds for newlines, so your NR signature cannot include any carriage return/line feed (CR/LF) pairs. Your non-PeopleSoft application must strip out the CRs before inserting the signature and sending the message.

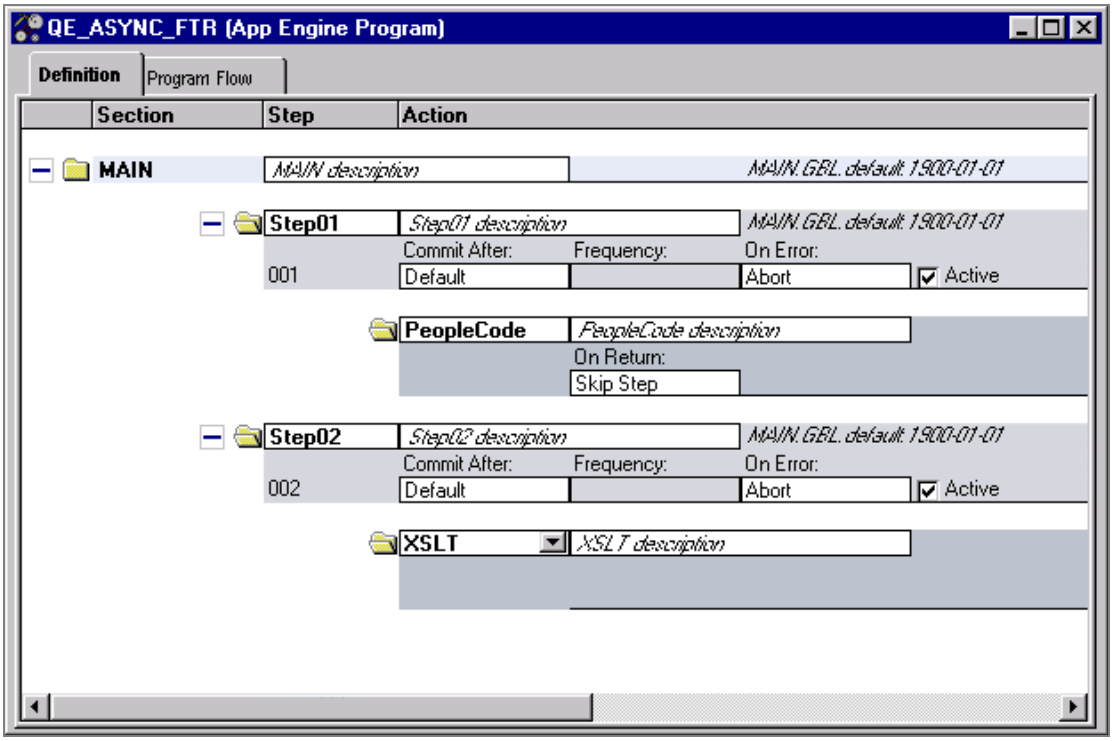
Developing Transform Programs

PeopleSoft's Application Engine provides the framework for Integration Broker transformation, translation and filtering. To use this framework, you create an Application Engine program of type *Transform Only*.

This section explains how to:

- Define a transform program.
- Work with transform programs.
- Access message data.
- Make working data available globally.

Defining a Transform Program



Application Engine transform program definition

To define a transform program:

1. In Application Designer, create a new Application Engine program.

   Select File, New, App Engine Program.

2. Open the Program Properties and select the Advanced tab.

3. Select a program type of *Transform Only*, click OK and save the program.

4. Insert sections, steps and actions as needed.

   Construct your program the same way as any other Application Engine program, using a combination of actions of type *XSLT* and *PeopleCode.*

See Also

Filtering Messages and Generating Errors

Applying Transformations

Performing Data Translation

PeopleTools PeopleBook: Application Engine, "Introducing Application Engine"

Working with Transform Programs

Integration Broker supports the use of Extensible Stylesheet Language Transformation (XSLT) code and PeopleCode, implemented in your transform program as one or more actions of type *XSLT* or *PeopleCode,* respectively. Following are some points to keep in mind:

- Each transform program step operates on the message content that results from the previous step, so you can break your transform program into a sequence of discrete steps.
- Multiple transform actions within a step can produce unwanted effects, so be sure to insert each XSLT or PeopleCode action in its own step.
- XSLT works only on XML DOM compliant data, so Integration Broker assures that both outbound and inbound messages are in XML DOM compliant form when transform programs are applied to them.

A transformation can modify an entire message until it no longer resembles the original at all. So can a data translation. The difference is that you must hard code everything you want to accomplish in a transformation, whereas the data translation relies on a repository of *codeset* metadata that you define. This means you can establish consistent rule-based translations, and reuse the same data without having to re-enter it.

You can combine transformation and data translation in a single transform step. We recommend you keep these processes in separate steps if possible, producing a modular program you can more easily maintain, and code you can reuse in other transform programs.

Deciding Which Language to Use

- XSLT is a well-recognized standard language perfectly suited to manipulating XML structures, so it's highly recommended for implementing transformations.
- Because of its straightforward template-based approach to accessing the codeset repository, XSLT is highly recommended for data translation.
- Currently, the ability to filter messages based on content is not available through XSLT, so filtering must be implemented in PeopleCode.
- You can use both XSLT and PeopleCode steps in a single transform program.

Each XSLT program must be enclosed in the following wrapper:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
   ...
</xsl:stylesheet>
```

Third party XSLT development tools may generate a wrapper that specifies a different URI. Make sure the URI in your program is exactly as shown here, or your program may not work as expected.

Note. You can find more information about XSLT at the World Wide Web Consortium (W3C) website. See http://www.w3.org/Style/XSL/.

Invoking a Transform Program

Your transform program is invoked by Integration Broker if you specify its name in the Transformations group Request or Response field when you edit the details of a transaction modifier. See "Administering Relationships," Configuring a Relationship for more information.

Tracing a Transform Program

For debugging purposes, you can trigger a trace of your transform program. Do this by adding a specific value to the Application Engine *trace* parameter, in one of the following ways:

- Specify the TRACE switch on the Application Engine command line, with the value ***8192*** added, for example:

```
-TRACE 8192
```

- Add the value ***8192*** to the TRACEAE parameter in the appropriate application server or Process Scheduler server configuration file, for example:

```
TRACEAE=8192
```

See Also

PeopleTools PeopleBook: PeopleCode Reference, "XmlDoc Class"

PeopleTools PeopleBook: Application Engine, "Tracing Application Engine Programs"

PeopleTools PeopleBook: Process Scheduler, "Appendix D: Using the PSADMIN Utility"

Accessing Message Data

When Integration Broker invokes a transform program, it inserts the message content into a PeopleCode system variable, **%TransformData**, which remains in scope throughout the program. Each step can access the variable in turn, modifying its content, which then becomes available to the next step.

XSLT and PeopleCode steps access %TransformData differently:

- In XSLT, the data is automatically made available to your program. The XSLT program is literally a presentation of the output structure and data, which includes **xsl** tags that reference, process and incorporate the input data into the output structure. There's no need to explicitly refer to %TransformData, which automatically receives the interpreted result of the XSLT.

- In PeopleCode, you must use the PeopleCode **TransformData** class to access %TransformData. You then access the XML data as a property of the TransformData object called **XmlDoc**, which you assign to an XmlDoc object and process normally. Because the XmlDoc object is a reference to the data portion of %TransformData, your modifications are automatically passed back to the system variable.

Using the TransformData Class

The PeopleCode TransformData class has several properties:

XmlDoc	Contains the XML message data. You can assign this to an XmlDoc object and process the data using the XmlDoc class methods and properties. This property is read/write.
Status	Set to **0** for success, the default value. Set to **1** to indicate the message failed a filtering step. Set to **2** to indicate an error occurred. This property is read/write.
SourceNode	The name of the source node. This property is read only.
DestNode	The name of the destination node. This property is read only.
SourceMsgName	The name of the source message. This property is read only.
DestMsgName	The name of the destination message. This property is read only.
SourceMsgVersion	The name of the source message version. This property is read only.
DestMsgVersion	The name of the destination message version. This property is read only.

The XmlDoc property is the key to accessing message data in a PeopleCode step; the Status property provides the means of communicating the success or failure of the transform program step to Integration Broker. See Filtering Messages and Generating Errors for details and examples.

Working with Message Samples

When developing a transform program for XML message data, you may find it useful to have sample copies of the initial and resulting XML message structures as a guide. You can generate the samples from the message definitions using the Create Test Message feature of Application Designer. See “Defining Messages,” Generating a Test Message.

After you publish a test message containing sample data, you can view its XML structure and content in Integration Broker Monitor, and copy that data to a text file you can use for reference. See “Using Integration Broker Monitor.”

Handling Non-XML Data

Because they work only with XML DOM compliant data, neither XSLT nor PeopleCode transform steps can process non-XML data. The XML DOM provides a way to incorporate such data into an XML structure so your transform programs won't produce errors.

If you're generating a non-XML outbound message in your PeopleSoft application, it's up to you to insert your message content into a special **xml** section containing a **CDATA** tag, as follows:

```
<xml psnonxml="yes">
   <![CDATA[nonXML_message_data]]>
</xml>
```

The following restrictions apply to the content of inbound non-XML messages, like CSV or PDF format, sent by third party applications:

- Inbound non-XML text messages must be encoded as UTF-8 compliant characters.
- Inbound non-text, or binary, messages must be encoded in base64 format.

Integration Broker provides for non-XML messages by automatically inserting the entire message content into an **xml/CDATA** wrapper upon receiving the message.

Making Working Data Available Globally

Currently, XSLT transform steps can't access external data, but PeopleCode can. XSLT also has no global variables, but the message itself is global, and can be used to pass working or external data to all steps in the transform program. During a PeopleCode step you can add a special node to the message to contain the data, which is then available to all subsequent transform steps.

Following is an example of a minimal input message:

```
<?xml version="1.0"?>
<Header>
  <LANGUAGE_CODE>en_us</LANGUAGE_CODE>
  <STATUS_CODE>1000</STATUS_CODE>
</Header>
```

The following PeopleCode inserts a node in the message to contain working data, by convention called **psft_workingstorage.** Then the PeopleCode inserts the current system date into that node:

```
/* Get the data from the AE Runtime */
Local TransformData &incomingData = %TransformData;
/* Set a temp object to contain the incoming document */
Local XmlDoc &inputDoc = &incomingData.XmlDoc;

/* Add a working storage node*/
```

```
Local XmlNode &wrkStorageNode =
&inputDoc.DocumentElement.AddElement("psft_workingstorage");

/* Add the current system date to the working storage*/
Local XmlNode &sysDateNode = &wrkStorageNode.AddElement("sysdate");
&sysDateNode.NodeValue = String(%Date);
```

Following is the resulting output message:

```
<?xml version="1.0"?>
<Header>
  <LANGUAGE_CODE>en_us</LANGUAGE_CODE>
  <STATUS_CODE>0</STATUS_CODE>
  <psft_workingstorage>
    <sysdate>2002-01-24</sysdate>
  </psft_workingstorage>
</Header>
```

Any subsequent transform step now has access to the current system date. Make sure the last step that uses the **psft_workingstorage** node removes it from the final output, as with this XSLT fragment:

```
<xsl:template match="psft_workingstorage">
   <!-- Do not copy this node -->
</xsl:template>
```

See Also

http://www.w3.org/Style/XSL/

Filtering Messages and Generating Errors

Integration Broker's filtering feature enables you to suppress an input message based on its content. For example, you can suppress all inbound purchase order messages that specify order quantities less than the minimum number required for a discount.

It's a good idea to place filtering steps early in your Application Engine transform program; each message suppressed by the filter is one less message for subsequent steps to process.

Currently, you must use PeopleCode for filtering, so it'll probably be a distinct step (XSLT is highly recommended for transformation and data translation). Because you must use the XmlDoc and XmlNode classes in your PeopleCode transform steps, you can analyze messages in any way that those classes support.

The Filtering Process

Filtering requires the following actions in your PeopleCode program:

1. Retrieve the message content from the %TransformData system variable.

2. Examine the desired criteria for which you're filtering the message.

3. If the message meets your criteria, do nothing further. It remains intact in the %TransformData system variable for the next step to process.

4. If the message fails to meet your criteria, replace the entire message content with a single node called **Filter,** containing the reason it failed:

```
<?xml version="1.0"?>
<Filter>reason_for_failure</Filter>
```

5. Set the TransformData Status property to **1** to indicate failure.

Integration Broker examines the Status property after each step, aborts the transform program if its value is **1.** You can then view the message in Integration Broker Monitor and see the reason for the failure.

Working with a PeopleCode Filtering Example

The following example of filtering presents an example input message, the PeopleCode filtering program, and the resulting output message.

Input Message

This is the input to the filtering step. Notice the line item order quantities (shown in bold):

```
<?xml version="1.0"?>
<PurchaseOrder>
   <Destination>
      <Address>123 Vine Street</Address>
      <Contact>
         <Name>Joe Smith</Name>
      </Contact>
      <Delivery type="ground">
         <Business>FedEx</Business>
      </Delivery>
   </Destination>
   <Payment>
      <CreditCard cardtype="visa">4024-9920-9892-8982</CreditCard>
   </Payment>
   <LineItems count="2">
      <Li locale="en_us" number="1">
         <Quantity>4</Quantity>
         <ProductName>pencil</ProductName>
         <UOM>box</UOM>
      </Li>
      <Li locale="en_us" number="2">
         <Quantity>10</Quantity>
```

```
         <ProductName>paper</ProductName>
         <UOM>large box</UOM>
      </Li>
   </LineItems>
</PurchaseOrder>
```

Note. Although this input message isn't in the PeopleSoft base message format, it is valid XML.

PeopleCode Filtering Program

This filtering program examines the line item order quantities of the input message, and generates the output message that follows:

```
/* Get the data from the AE Runtime */
Local TransformData &tempData = %TransformData;
/* Set a temp object to contain the incoming document */
Local XmlDoc &tempDoc = &tempData.XmlDoc;

/* Find the line items quantities contained in the incoming Purchase Order */
Local array of XmlNode &quantities =
&tempDoc.DocumentElement.FindNodes("LineItems/Li/Quantity");
/* Temp storage of a node */
Local XmlNode &tempNode;

/* Loop through the quantities and make sure they are all above 5 */
For &i = 1 To &quantities.Len
   /* Set the temp node*/
   &tempNode = &quantities [&i];
   /* Make sure the node isn't empty*/
   If ( Not &tempNode.IsNull) Then

      /* Check the value, if not greater than 5 this does not pass filter*/
      If (Value(&tempNode.NodeValue) < 5) Then
         /* Clear out the doc and put in the "Filter" root node */
         If (&tempDoc.ParseXmlString("<?xml version=""1.0""?><Filter/>")) Then
            /* Get the new root node and set the value to be the reason for
failing filter */
            &rootNode = &tempDoc.DocumentElement;
            &rootNode.NodeValue = "Line item quantity was found that was less
than 5!";
            /* Set the status of the transformation to 1 for failed filter*/
            &tempData.Status = 1;
         End-If;
         Break;
      End-If
```

```
   End-If
End-For;
```

Output Message

This is the result of applying the PeopleCode filtering program:

```
<?xml version="1.0"?>
<Filter>Line item quantity was found that was less than 5!</Filter>
```

Generating an Error

You may have reasons to abort a transform program that aren't considered error conditions by Integration Broker. In PeopleCode steps, you can force the transform program to abort, and generate a readable error message as well:

1. Replace the entire message content with a single node called **Error,** containing the reason for the error:

```
<?xml version="1.0"?>
<Error>reason_for_error</Error>
```

2. Set the TransformData Status property to **2** to indicate error status.

Integration Broker examines the Status property after each step, aborts the transform program if its value is **2.** You can then view the message in Integration Broker Monitor and see the reason for the error.

Note. If an XSLT or PeopleCode step fails, Integration Broker automatically sets the Status property to ***2*** and aborts the transform program, but you can't provide your own error message.

See Also

"Using Integration Broker Monitor"

Applying Transformations

A transformation may be needed when one node is sending a request or response message with a data structure different from the structure required by the other node. Either or both of the participating nodes are PeopleSoft applications. At either end of the transaction, any of the following structure types may be required:

- The PeopleSoft standard base message format. This is the rowset structure most typical for PeopleSoft applications, which is XML DOM compliant. See "Sending and Receiving Messages," PeopleSoft Rowsets for more information.

- An XML DOM compliant non-rowset based structure. This is generic XML data. See "Sending and Receiving Messages," XML Document Object Model for more information.
- A SOAP compliant XML structure. This is also XML DOM compliant. See "Sending and Receiving Messages," Simple Object Access Protocol for more information.
- A non-XML structure. Third party applications are more likely than PeopleSoft applications to require this type.

Your transformation can be between different structure types, or between different structures of the same type.

Using XSLT for Transformation

An XSLT transformation simulates the original message structure, then specifies how to treat nodes within that structure. You can:

- Copy the original content of a node without changing anything.
- Define and insert a new version of a node.
- Enter any structure or content directly.
- Eliminate a node by omitting reference to it, or by not inserting anything new in its place.

An XSLT Transformation Example

The XSLT wrapper is required:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
```

The primary node tag matches the original message structure by matching its top level content tag, the message name. Between the template tags, you can insert any structure or content you want. Integration Broker replaces each *xsl* tag with the data it references, producing a transformed message as the output of the step.

```
    <xsl:template match="QE_SYNC_MSG">
       <QE_SYNC_MSG>
          <xsl:copy-of select="FieldTypes"/>
          <MsgData>
             <Transaction>
                <xsl:apply-templates
select="MsgData/Transaction/QE_SALES_ORDER"/>
                <xsl:copy-of select="MsgData/Transaction/PSCAMA"/>
             </Transaction>
```

```
        </MsgData>
      </QE_SYNC_MSG>
    </xsl:template>
```

The following node is defined to match a record in the input message by its top level content tag, the record name. This template is applied by the *xsl:apply-templates* tag of the preceding node (shown in bold).

Between the template tags, you can insert any structure or content you want. In this example, **90** is prepended to the QE_ACCT_ID value, and the QE_ACCOUNT_NAME field is renamed to QE_ACCOUNT (shown in bold). Also, any existing value in the DESCRLONG field is removed, and the remaining fields are passed through with their original values.

```
    <xsl:template match="QE_SALES_ORDER">
       <QE_SALES_ORDER><xsl:attribute name="class"><xsl:value-of
select="@class"/></xsl:attribute>
         <xsl:variable name="temp" select="QE_ACCT_ID"/>
         <QE_ACCT_ID><xsl:value-of select="concat(90,$temp)"/></QE_ACCT_ID>
         <QE_ACCOUNT><xsl:value-of select="QE_ACCOUNT_NAME"/></QE_ACCOUNT>
         <QE_ADDRESS><xsl:value-of select="QE_ADDRESS"/></QE_ADDRESS>
         <QE_PHONE><xsl:value-of select="QE_PHONE"/> </QE_PHONE>
         <QE_FROMROWSET/>
         <QE_TOROWSET/>
         <QE_SEND_SOA_BTN/>
         <QE_SEND_SOS_BTN/>
         <DESCRLONG></DESCRLONG>
       </QE_SALES_ORDER>
    </xsl:template>
```

Finally, you need the closing wrapper tag:

```
</xsl:stylesheet>
```

Note. You can find more information about XSLT at the World Wide Web Consortium (W3C) website. See http://www.w3.org/Style/XSL/.

Performing Data Translation

This section explains how to:

- Define a codeset group.
- Define a codeset.
- Define codeset values.
- Use XSLT for data translation.

- Work with an XSLT translation example.

Understanding Data Translation

Data translation is best suited for modifying message content rather than structure, although you can also make local structural changes. It's most appropriate when the sending and receiving systems use different field values, or different combinations of fields and their values, to represent the same information.

A Sample Scenario

- Application A transmits customer names in four fields — **Title, First, Middle, Last.**
- Application B uses two fields — **Last, First.** It doesn't use a title, but includes the middle name as part of the *First* field.
- Application C uses only one field — **AccountID.**

Clearly, the representation used by one application won't be understood by either of the other two. Integration Broker can apply a transform program to translate each of these representations into a version appropriate to the target application.

One Integration Broker node can store in its codeset repository the equivalent fields and values used by another node. When it receives a message from the other node containing a customer name, it can use its codeset repository to translate the information into the form it prefers. It can likewise reverse the process for messages it sends to the other node.

For a given integration, your circumstances and preferences determine how you allocate the responsibility for performing data translation. You can distribute the translation activity among the participating nodes, or you can designate one Integration Broker node to do all the data translation, whether the messages are inbound, outbound, or being redirected between the other nodes. Using a single node, if possible, can reduce the need for duplicating repository data.

The Elements of a Data Translation

The following elements constitute the codeset repository, managed as PIA components:

Codeset group

Maintains a list of the significant data fields and their values that a particular node might send in an initial message. These are name/value pairs a translation program might find (match) and use as the basis for determining what the result message should contain. These name/value pairs are known as *match names* and *match values.*

Each Integration Broker node requiring data translation must belong to a codeset group. See Defining a Codeset Group.

Codeset	A specific set of match name/match value pairs selected from an existing codeset group. The selected name/value pairs are the basis for possible field value combinations you wish to match in a message, and to which your translation program can respond by modifying the message content. Each codeset typically represents one set of fields among possibly many requiring translation for a given message. See Defining a Codeset.
Codeset values	A codeset value is a named value you predefine, also known as a *return value.* Your translation program can output the return value as a result of matching a specific combination of match values. You associate multiple combinations of codeset values with the combination of a source codeset group, a codeset from that source group, and a target codeset group. For each permutation of match values selected from the codeset, you define a different combination of codeset values to apply to your result message. See Defining Codeset Values.

The other key element of data translation is your translation program, which invokes the codesets and codeset values you've defined.

The Data Translation Development Sequence

You must initially define these elements in a particular order:

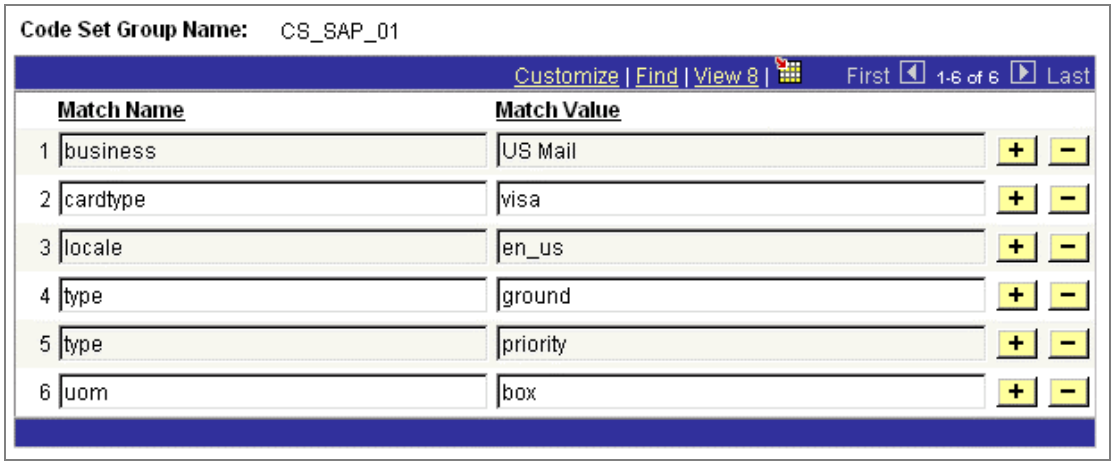
1. The codeset group must exist before you can define a codeset based on it.

2. A codeset and two codeset groups must exist so you can define codeset values associated with them.

3. A codeset and associated codeset values must exist before you can invoke them in your translation program.

However, it's unlikely that you'll be able to fully define any of these elements without some trial and error. You may find you'll have to modify and extend each element in turn as you develop your data translation. Keep this in mind as you read about each element in the nominal sequence presented in this section.

Defining a Codeset Group



Codeset group page

To define a codeset group:

1. Select PeopleTools, Integration Broker, Codeset Groups.

   The Codeset Groups search page appears.

2. Add a new value, enter a codeset group name for your new group and click **Add.**

   Enter a name that reflects a common quality of the nodes you plan to assign to this group, for example, the name of the software they all use to manage shipping. The Codeset Groups component appears. This component consists of one page, on which you maintain a list of name/value pairs.

   Note. If you want to configure an existing codeset group, enter its name on the search page.

3. Click the **Add a new row** button in one of the rows.

   A new empty row appears below the row you clicked.

   Note. If the codeset group has no name/value pairs entered, an empty row will already be available.

4. Enter a match name.

   This is the name of a data field that might be part of a message sent by a node belonging to this codeset group. You don't have to create an entry for every field, just the ones that you'll need to translate, or use for reference in a translation.

5. Enter a match value.

This is one of the possible values of the data field that might be part of a message sent by a node assigned to this codeset group.

6. Repeat steps **3** through **5** for each of the significant name/value pairs you expect to appear in a message.

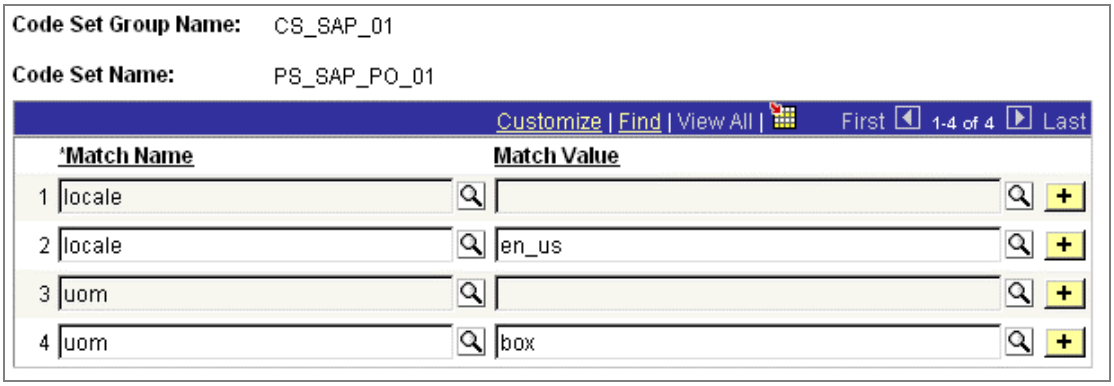
   This doesn't have to be all possible values of all of the message fields, just the names and values you expect to require translation.

7. Assign one or more nodes to this codeset group.

   Every source and destination node involved in a data translation must belong to a codeset group. You must assign each participating node to an appropriate codeset group by an entry in its node properties.

   The assignment for each node is required only in the database of the node performing the data translation. This translating node needn't be either the source or the target. Multiple nodes that represent data the same way may be assigned to the the same codeset group. See "Administering Basic Integrations," Specifying General Node Information.

Defining a Codeset



Codeset page

To define a codeset:

1. Select PeopleTools, Integration Broker, Codesets.

   The Codesets search page appears.

2. Add a new value, and enter a codeset group name on which you want this codeset to be based.

3. Enter a codeset name and click **Add.**

   Enter a name that reflects the purpose of this codeset, for example, to translate the representation of a shippping method in a message. The Codesets component appears. This component consists of one page, on which you enter a list of name/value pairs.

Note. If you want to configure an existing codeset, enter its name or the name of its associated codeset group on the search page.

4. Click the **Add a new row** button in one of the rows.

   A new empty row appears below the row you clicked.

   Note. If the codeset has no name/value pairs entered, an empty row will already be available.

5. Select a match name from the set of match names defined for the associated codeset group.

   This is the name of a field containing a value you need to match for the purposes of this codeset.

6. Select a match value from the set of match values defined for the selected match name.

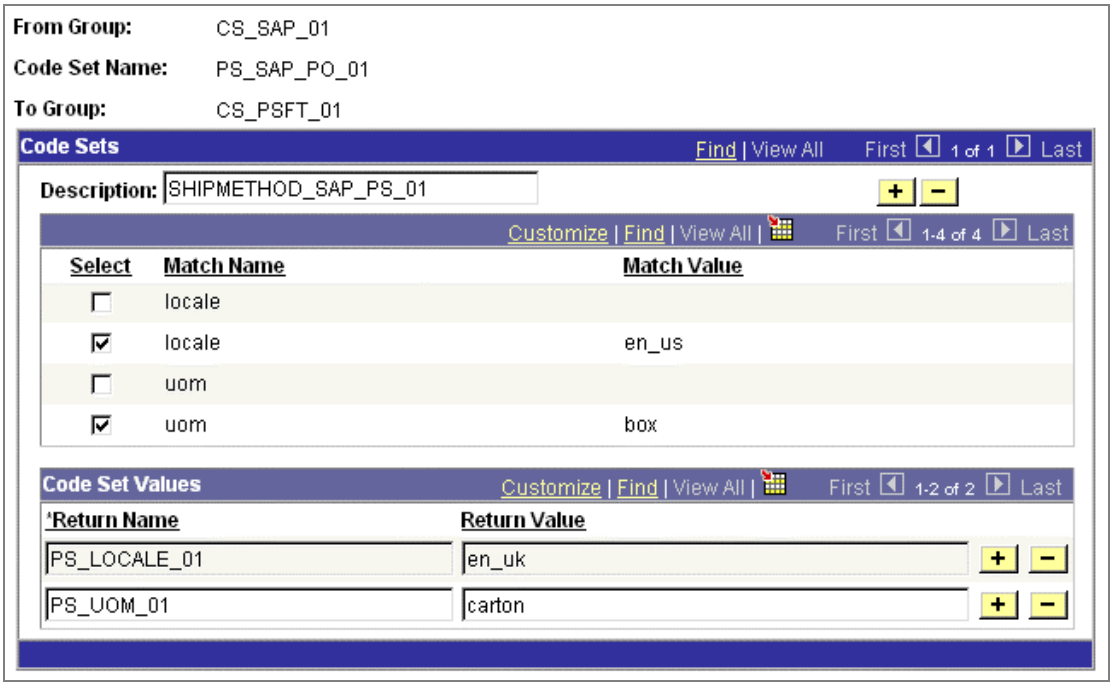
   This is the value of the field that your translation program needs to match so it can initiate a translation in response to that value.

   Note. You can leave the value blank. If so, you should do the same for each match name in this codeset, in addition to any other values you select for them. A combination consisting of all blank values is treated as a "wild card" by Integration Broker, which enables it to respond to unanticipated values specified in your translation program with default behavior that you define.

7. Repeat steps **4** through **6** to enter all the name/value pairs that may need to be matched.

   The name/value pairs you select should encompass only the possible value combinations that your translation program needs to match for a single translation.

Defining Codeset Values



Codeset values page

To define codeset values:

1. Select PeopleTools, Integration Broker, Codeset Values.

   The Codeset Values search page appears.

2. Add a new value, and select a codeset group name for the From Group.

   This is the codeset group to which the source node belongs.

3. Select a codeset name from the codesets based on the From Group you selected.

   This is the codeset whose match name/match value permutations you wish to match.

4. Select a codeset group name for the To Group.

   This is the codeset group to which the target node belongs.

5. Click **Add.**

   The Codeset Values component appears. This component consists of one page: The upper grid contains all of the selected codeset's match name/match value pairs, and the lower grid contains the return values you specify. Each permutation you define has its own Description field, which can help you distinguish between potentially numerous permutations that may be subtly different from each other.

Note. If you want to configure an existing codeset values definition, enter its From Group, Code Set Name and To Group keys on the search page.

6. Select check boxes to define a permutation of match name/match value pairs.

   For each match name, you can select at most one match value.

   Note. A permutation consisting of all blank values serves as a "wild card;" that is, it matches any input value combination that isn't matched by any other pemutation. However, a permutation with some blank and some non-blank values works differently; it requires the names with blank values to actually match blank field values in the input data.

7. In the Code Set Values grid, enter a return name and a return value for that name.

   You can use any return name you want, because only your codeset translation program refers to it. Your translation program can use the return value as a field value or as a node name in the output data.

8. (Optional) In the Code Set Values grid, click the **Add a New Row** button, and repeat step **6.**

   Add as many return name/return value pairs as you need for your output based on the current permutation. If the permutation is matched in the input data, the code set values you define for that permutation become available for you to call and insert in the output data.

9. (Optional) At the top level of this page, click the **Add a New Row** button, and repeat steps **5** through **7.**

   This inserts a new permutation row, in which you can define a different permutation of match name/match value pairs that you expect for the current codeset. For each permutation, you'll define a separate, independent set of codeset values.

Other Considerations

Each permutation has its own *Description* field, which can help you distiguish between potentially numerous permutations that may be subtly different from each other.

You'll generally define only permutations that you expect the input data to contain, but make sure you allow for unforseen match values by including permutations with blank values. You can then specify default return values for those permutations. With a large number of match names in the codeset, you can make sure to catch all unforseen combinations by defining a permutation with all blank match values.

Important! The set of return names you define *must be identical* for all of the permutations of match name/match value pairs for the current codeset in this definition. Your translation program invokes the codeset and applies the return names from this definition, but it can't anticipate which permutations will be matched, or which actual return values it's applying — just the return names.

Using XSLT for Data Translation

Once you've defined the match name/match value permutations for a codeset with respect to a given target codeset group, and defined the return values for those permutations, you can write an XSLT translation program that invokes that codeset and applies the return values.

An XSLT translation is based on XSLT transformation structure — in fact, you could combine both tasks into a single program, but it's recommended that you keep them separate for easier understanding and maintenance. See Using XSLT for Transformation for information about basic XSLT program structure.

The psft_function Node

To implement data translation capability, Integration Broker provides an enhancement to the standard XSLT model in the form of a tag called **psft_function**. Each **psft_function** node in your program comprises a single instance of data translation that invokes a particular codeset and applies a specified set of codeset values.

Note. You can insert a **psft_function** node anywhere inside the template containing the fields you want to translate. However, you'll find it easiest to place it at or near the point in the template where the return values will go, to avoid having to specify a complex path to that location.

The **psft_function** tag has the following attributes:

name="codeset"	The value of this attribute must be **codeset.**
codesetname	The name of the codeset whose name/value permutations you want to match in the input data. The transaction modifier that invokes this transform program identifies the source and target nodes involved, and Integration Broker examines their definitions to determine the From Group and To Group. The combination of these two keys and the codeset name identifies the codeset values definition to apply.
source	(Optional) Use this attribute to override the name of the source node specified by the transaction modifier. Integration Broker uses the specified node's codeset group as the From Group key, thus invoking a different codeset values definition.
dest	(Optional) Use this attribute to override the name of the target node specified by the transaction modifier. Integration Broker uses the specified node's codeset group as the To Group key, thus invoking a different codeset values definition.

Note. The **source** and **dest** attributes don't change the source or target nodes; they just invoke the codeset groups to which those nodes belong.

Following is an example of **psft_function** using all of its attributes:

```
<psft_function name="codeset" codesetname="PS_SAP_PO_01" source="SAP_02"
dest="PSFT_03">...</psft_function>
```

The parm and value Tags

The **psft_function** node can contain two tags, **parm** and **value:**

- **parm —** Use this tag to specify a match name from the codeset values definition you specified for this translation. You do this with the tag's only attribute: **name.** Set this to a match name from the codeset values definition.
  - The **parm** tag should contain a match value, usually specified as an **xsl:value-of** tag that identifies where the value resides in the input data. See http://www.w3.org/Style/XSL/ for details about the **xsl:value-of** tag.
  - Use one **parm** tag for each distinct match name in the codeset values definition.
- **value —**Use this tag to specify a return name from the codeset values definition you specified for this translation, and identify where to place the return value assigned to that return name for the matched permutation, and how to apply that value.
  - Use one **value** tag for each return name in the codeset values definition that you want to use in your output. The **value** tag has the following attributes:

name	A return name from the codeset values definition you specified for this translation. The return value assigned to this return name can be used as a data value or as a node name in your output depending on the other attributes you specify.
select	An XSLT path (XPATH) to the location where the return value should be applied in the output data. See http://www.w3.org/Style/XSL/ for details about XPATH.
createIfDNE	(Optional) Set this attribute to **yes,** or don't include the attribute (meaning **no,** the default). *Yes* ensures that the node specified by the **select** attribute will be created if it Does Not Exist yet. The return value is inserted as the value of that node.
createNodeFromValue	(Optional) Set this attribute to **yes,** or don't include the attribute (meaning **no,** the default). If it's **yes,** the return value will be used as the name of a new node, created where the **select** attribute specifies. In this case, the **value** tag can contain a valid XSLT value for that node, usually specified as an **xsl:value-of** tag that identifies where the value resides in the input data. See http://www.w3.org/Style/XSL/ for details about the **xsl:value-of** tag.

Following is an example of a **value** field:

```
<value name="PS_RET_01" select="." createNodeFromValue="yes"><xsl:value-of
select="CreditCard"/></value>
```

Working with an XSLT Translation Example

The following example of XSLT data translation presents an example input message, the XSLT translation program, and the resulting output message.

Input Message

This is the input to the XSLT translation:

```
<PurchaseOrder>
   <Destination>
      <Address>123 Vine Street</Address>
      <Contact>
         <Name>Joe Smith</Name>
      </Contact>
      <Delivery type="ground">
         <Business>FedEx</Business>
      </Delivery>
   </Destination>
   <Payment>
      <CreditCard cardtype="visa">4024-9920-9892-8982</CreditCard>
   </Payment>
   <LineItems count="2">
      <Li locale="en_us" number="1">
         <Quantity>15</Quantity>
         <ProductName>pencil</ProductName>
         <UOM>box</UOM>
      </Li>
      <Li locale="en_us" number="2">
         <Quantity>10</Quantity>
         <ProductName>paper</ProductName>
         <UOM>large box</UOM>
      </Li>
   </LineItems>
</PurchaseOrder>
```

Note. Although this input message isn't in the PeopleSoft base message format, it is valid XML.

XSLT Data Translation Program

This translation program processes the input message in this example, and generates the output message that follows. The statements shown in bold demonstrate some uses of the **psft_function** node:

```
<?xml version="1.0"?>
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
version="1.0">
   <xsl:template match="PurchaseOrder">
      <po>
         <xsl:apply-templates/>
      </po>
   </xsl:template>

   <xsl:template match="Destination">
      <dest>
         <address><xsl:value-of select="Address"/></address>
         <name><xsl:value-of select="Contact/Name"/></name>
         <delivery>
            <type>
               <psft_function name="codeset" codesetname="PS_SAP_PO_03"
dest="PSFT_03">
                  <parm name="type"><xsl:value-of
select="Delivery/@type"/></parm>
                  <value name="PS_RET_01" select="."/>
               </psft_function>
            </type>
            <carrier>
               <psft_function name="codeset" codesetname="PS_SAP_PO_03"
source="SAP_03">
                  <parm name="Business"><xsl:value-of
select="Delivery/Business"/></parm>
                  <value name="PS_RET_01" select="."/>
               </psft_function>
            </carrier>
         </delivery>
      </dest>
   </xsl:template>

   <xsl:template match="Payment">
      <payment>
         <psft_function name="codeset" codesetname="PS_SAP_PO_02">
             <parm name="cardtype"><xsl:value-of
select="CreditCard/@cardtype"/></parm>
             <value name="PS_RET_01" select="."
createNodeFromValue="yes"><xsl:value-of select="CreditCard"/></value>
         </psft_function>
      </payment>
```

```
   </xsl:template>

   <xsl:template match="Li">
      <li><xsl:attribute name="id"><xsl:value-of
select="@number"/></xsl:attribute>
         <name><xsl:value-of select="ProductName"/></name>
         <qty><xsl:value-of select="Quantity"/></qty>
         <uom>
            <psft_function name="codeset" codesetname="PS_SAP_PO_01">
                <parm name="locale"><xsl:value-of select="@locale"/></parm>
                <parm name="uom"><xsl:value-of select="UOM"/></parm>
                <value name="PS_RET_01" select="."/>
                <value name="PS_RET_02" select="../type" createIfDNE="yes"/>
            </psft_function>
         </uom>
      </li>
   </xsl:template>
</xsl:stylesheet>
```

Output Message

This is the result of applying the XSLT translation:

```
<po>
   <li id="1">
      <name>pencil</name>
      <qty>15</qty>
      <uom>Carton</uom>
      <type>Bic</type>
   </li>
   <li id="2">
      <name>paper</name>
      <qty>10</qty>
      <uom>Box</uom>
      <type>Bic</type>
   </li>
   <dest>
      <address>123 Vine Street</address>
      <name>Joe Smith</name>
      <delivery>
         <type>Ground</type>
         <carrier>Federal Express</carrier>
      </delivery>
   </dest>
   <payment>
      <VISA>4024-9920-9892-8982</VISA>
   </payment>
```

```
</po>
```

CHAPTER 11

Administering Relationships

This chapter explains how to:

- Determine relationship parameters.
- Configure a relationship.
- Manage transaction modifiers.

Understanding Relationships

Every integration requires at least one transaction at each Integration Broker node: One node uses a transaction to send a message, and one or more nodes use transactions to receive the message. The sending node may apply a transaction with different parameters than the nodes that ultimately receive the message — with respect to routing, transmission type, message structure, or message content. A *relationship* reconciles these incompatible parameters, producing a successful transmission of data from the source to the destination.

When to Use Relationships

You must define a relationship if any of the following is true:

- The sending and receiving systems use different transmission types for an integration point.
- You're using a hub configuration to route messages.
- One or more messages in a transaction needs to be transformed, translated or filtered upon sending or receiving.

Relationships *aren't* required for basic messaging, if all of the following are true:

- The transactions defined at each participating node specify the same parameters (except transmission direction, which depends on a node's point of reference).
- You're not using a hub configuration (messages are sent directly from the source node to the target nodes).
- Each node expects the message to have the same structure, version and encoding as the other nodes.

Note. As you work with relationships, keep in mind that the relationship definitions you create, and the node definitions, transaction definitions and transform programs they reference, are all stored in the database you're signed on to, which is the default local node.

See Also

"Applying Transformation, Translation and Filtering"

"Understanding Integrations"

Transaction Modifiers

A relationship applies a *transaction modifier* to an initial transaction, which produces a resulting transaction that's appropriate for the target node. The transaction modifiers you define must produce a combination of initial and resulting transaction types supported by Integration Broker.

Supported Transaction Type Combinations

The following transaction type combinations are supported by Integration Broker relationships:

InAsync to InSync	Redirect an inbound asynchronous request message to a synchronous request handler.
OutAsync to OutSync	Route an outbound asynchronous request message at a non-hub node to a destination node that supports only synchronous. **Note.** This combination isn't supported when both the initial and resulting transactions specify the default local node.
InAsync to OutSync	Route an inbound asynchronous request message at a hub node to a destination node that supports only synchronous.
InAsync to OutAsync	Route an inbound asynchronous request message at a hub node.
InSync to OutSync	Route an inbound synchronous request message at a hub node.
OutAsync to OutAsync	Apply a transform program to an outbound asynchronous request message at a non-hub node.
OutSync to OutSync	Apply a transform program to an outbound synchronous request message at a non-hub node.
InAsync to InAsync	Apply a transform program to an inbound asynchronous request message.
InSync to InSync	Apply a transform program to an inbound synchronous request message.

Note. An outbound to inbound modifier would essentially define a local integration, in which the default local node is both source and target. However, unlike cross-node messaging, you need to define only one transaction for a local integration, and Integration Broker automatically routes the message appropriately, so no transaction modifier is necessary. See "Administering Basic Integrations," Configuring Transactions.

See Also

"Understanding Integrations"

"Applying Transformation, Translation and Filtering"

Determining Relationship Parameters

This section explains how to:

- Select a transaction type combination.
- Determine where to define the relationship.

Selecting a Transaction Type Combination

Your choice of which transaction type combination to apply with a transaction modifier depends on the difference between what the source node sends and what the target node expects. The following scenarios can be handled using relationships:

- *Different transmission types:* The source node sends an asynchronous request message, but the target node expects a synchronous message. Choose from the following solutions:
  - The target node converts the inbound asynchronous message to synchronous, using an InAsync to InSync modifier.
  - The source node converts the outbound asynchronous message to synchronous before sending it, using an OutAsync to OutSync modifier.
  - A hub node receives the inbound asynchronous message, and converts it to an outbound synchronous message, using an InAsync to OutSync modifier.
- *Hub routing:* You're using the default local node as an Integration Broker hub, and it receives an inbound request message. Choose from the following solutions:
  - For an asynchronous message — the hub routes it to the target node, using an InAsync to OutAsync modifier.
  - For an asynchronous message — the hub converts it to an outbound synchronous message and routes it to the target node, using an InAsync to OutSync modifier.

  - For a synchronous message — the hub routes it to the target node, using an InSync to OutSync modifier.

- *Transformation or translation:* The target node expects a different request message or request message version than the source node sends. Choose from the following solutions:

  - The source node applies a transform program to the outbound asynchronous message, using an OutAsync to OutAsync modifier.

  - The source node applies a transform program to the outbound synchronous message, using an OutSync to OutSync modifier.

  - The target node applies a transform program to the inbound asynchronous message, using an InAsync to InAsync modifier.

  - The target node applies a transform program to the inbound synchronous message, using an InSync to InSync modifier.

Keep in mind that the source or target node for any of these scenarios can itself be a hub. Each transaction modifier applies to a single segment of the journey a message makes from the initial source node to the ultimate target node. Additionally, any transaction modifier can invoke a transform program.

Note. No special setup is required to use a hub configuration. Any node where you apply an inbound to outbound transaction modifier is by definition a hub node. You can treat any Integration Broker node simultaneously as a hub node and as a non-hub node for the same transaction. See Retaining Messages at a Hub Node.

See Also

"Applying Transformation, Translation and Filtering"

Determining Where to Define the Relationship

You define a relationship at the node that will change the transmission type, apply the routing or invoke the transform program specified by a transaction modifier. Your choice depends on which node is best suited for the solution you need to apply.

If you need more than one type of solution, you may be able to combine them in a single relationship, or you may find it necessary to define separate relationships at different nodes. You may have to consider which nodes you have control over, or whether you're using a hub configuration. In some cases, more than one approach may be available.

An example implementing all three solution types with three different relationships:

1. At the initial source node, change the transmission type using an OutAsync to OutSync modifier.

2. At the hub, route the message using an InSync to OutSync modifier.

3. At the ultimate target node, transform the message using an InSync to InSync modifier.

An example implementing all three solution types with a single relationship:

1. At the source node, use a single OutAsync to OutSync modifier to change the transmission type, route the message directly to a new target node, and transform the message.

Note. These three solution types can be applied with separate transaction modifiers at separate nodes, or they can be applied with a single transaction modifier at a single node, but only one transaction modifier is applied to a given message at any node. That is, multiple transaction modifiers can't be applied in sequence to a single message at a single node.

Configuring a Relationship

A relationship applies a transaction modifier, which modifies the transmission type, destination or message provided by the initial transaction, and uses the resulting transaction to pass the message on to its destination.

This section explains how to:

- Define a new relationship.
- Specify a node pair.
- Override node properties.
- Override contact information.

Defining a New Relationship

You configure relationships using the Relationships component, which consists of two pages. Use this component to specify two nodes, and assign a relationship ID to that node pair. You'll then define all the transaction modifiers that facilitate integrations between those two nodes, as part of the relationship definition.

Note. You can define only one relationship for each node pair.

To create a new relationship definition:

1. Select PeopleTools, Integration Broker, Relationships.

   The Relationships search page appears.

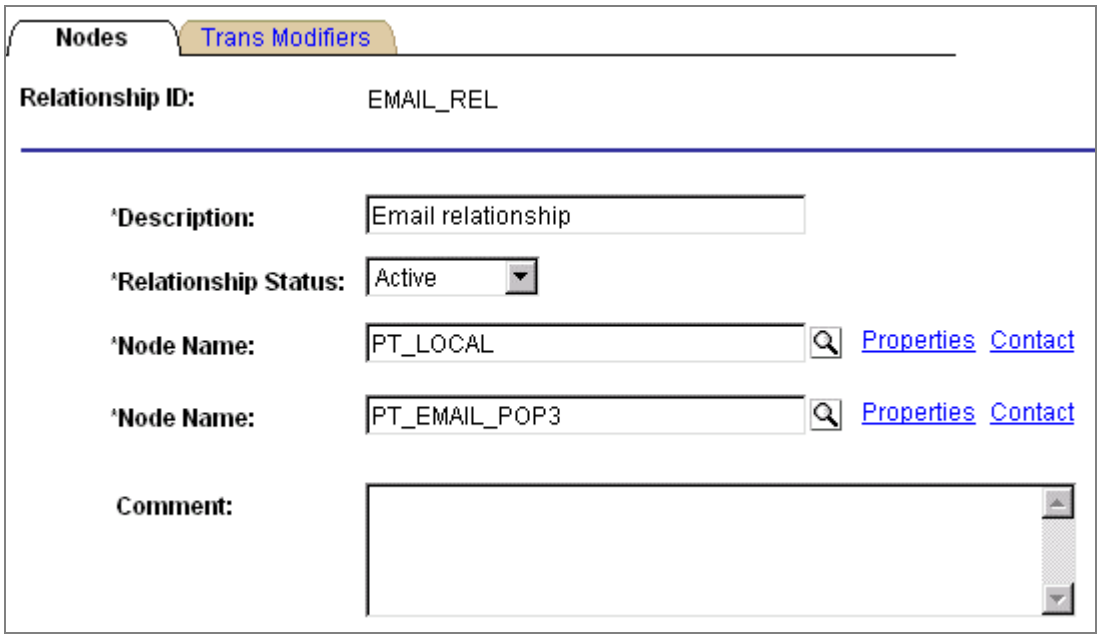
2. Add a new value, enter a relationship ID for your new relationship and click **Add.**

The Nodes page appears.

Note. If you want to configure an existing relationship, enter its relationship ID on the search page.

Specifying a Node Pair

Access the Relationships — Nodes page.



Relationships — Nodes page

Description Enter a more descriptive name for the relationship.

Relationship Status Select from the following:

Active: All of this relationship's transaction modifiers is eligible to be applied by the integration engine. This is the default value.

Inactive: None of this relationship's transaction modifiers is eligible to be applied by the integration engine.

Node Name Enter the names of the two nodes to which this relationship applies. Only transactions included in the selected nodes' definitions can be used by this relationship. You can specify the same node name twice, which still constitutes a pair as far as a relationship is concerned.

Note. Never enter the name of the default local node.

Properties	Click Properties to override the original properties of each node for this relationship. The Relationship Properties page appears. See Overriding Node Properties.
Contact	Click Contact to override the original contact information of each node for this relationship. The Relationship Contact page appears. See Overriding Contact Information.

Enter the appropriate node names on this page according to the following guidelines:

- For inbound to inbound transaction type combinations, enter the name of the source node in both fields.
- For outbound to outbound transaction type combinations, enter the name of the target node in both fields.
- For inbound to outbound (hub) transaction type combinations, enter the name of the source node for the inbound request message, and the name of the target node for the outbound request message. These must be different.

For any node pair you specify, the relationship can include transaction modifiers that send messages in either direction. Thus, if you initially define this relationship to handle an outbound to outbound combination, it also handles an inbound to inbound combination.

Activating Relationships

When a relationship is active, Integration Broker can immediately invoke it to apply its transaction modifiers to qualifying transactions. However, messaging depends on the status of the message and the node — their definitions in the local database must both be active as well. These elements have the following dependencies:

- When you deactivate a message, all associated relationships are deactivated.
- When you deactivate a node, all associated relationships are deactivated.
- You can specify only active nodes in a new relationship.
- You can activate any existing relationship without restriction.

Note. If you deactivate a node or a message, all relationships associated with it are also deactivated. Activating the node and the message won't automatically reactivate the relationships; they must be individually reactivated.

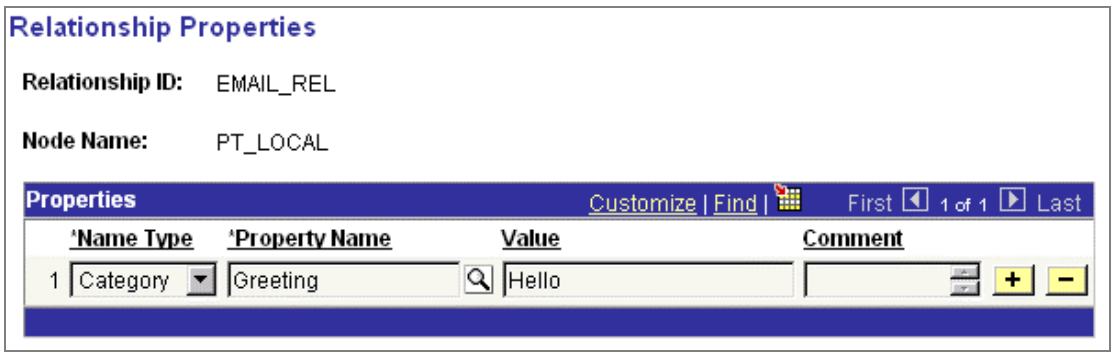
See Also

Editing Transaction Modifier Details

Overriding Node Properties

The Relationship Properties page provides a convenient place to store information about a node that's only meaningful in the context of a specific relationship. These properties can be used to update messages with additional information, such as adding a specific customer ID that's needed by your trading partner when you send orders. Properties created for all relationships are stored in a single table, PSRELATIONPROP.

The Relationship Properties page appears when you select Properties on the Relationships — Nodes page.



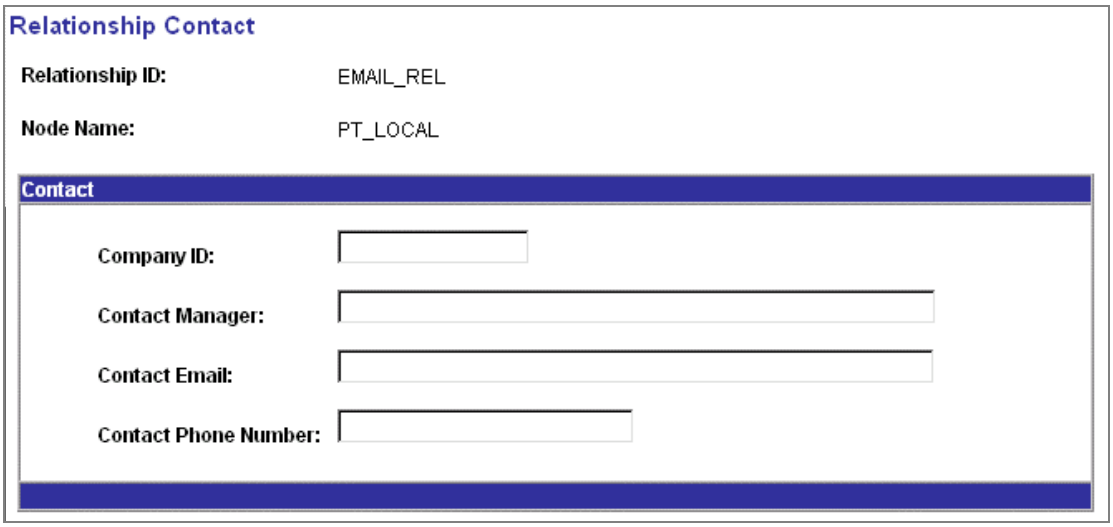
Relationship Properties page

Use this page to override the original properties of each node for this relationship. This page works the same way as the Node Definitions — Properties page. See "Administering Basic Integrations," Defining Node Properties.

Note. These properties are saved as part of the relationship. Be sure to save the relationship after you edit this page.

Overriding Contact Information

The Relationship Contact page appears when you select Contact on the Relationships — Nodes page.



Relationship Contact page

Use this page to override the original contact information of each node for this relationship. This page works the same way as the Node Definitions — Contact/Notes page. See "Administering Basic Integrations," Specifying Contact Information.

Note. This contact information is saved as part of the relationship. Be sure to save the relationship after you edit this page.

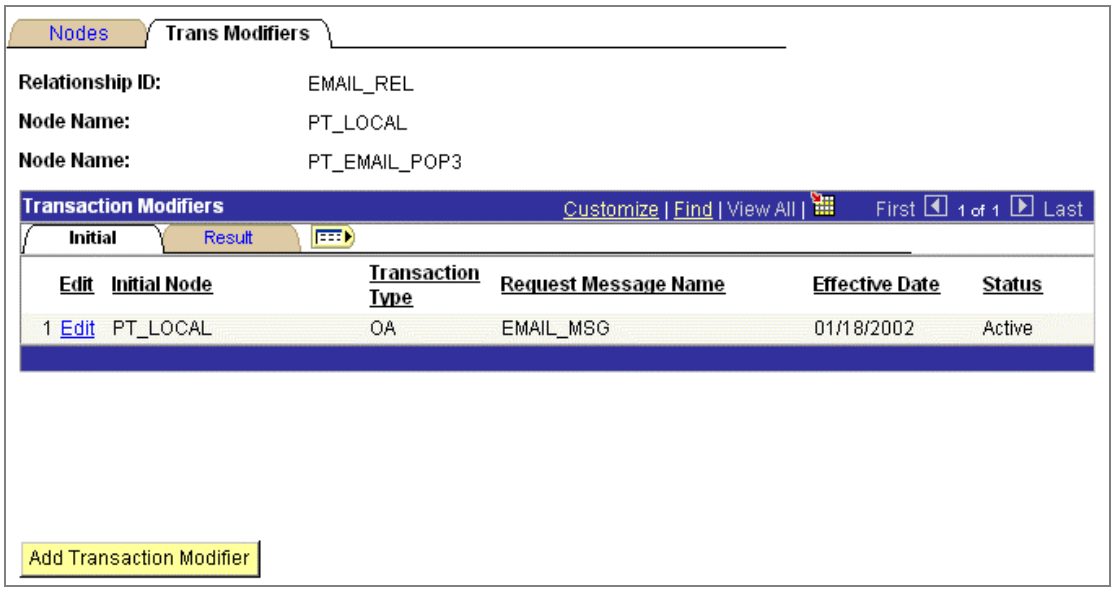
Managing Transaction Modifiers

This section describes how to:

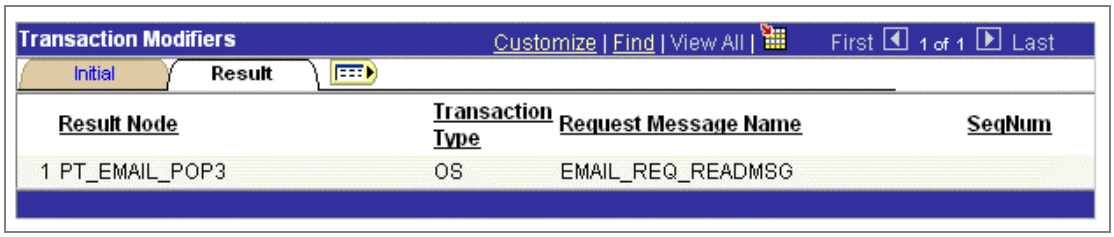
- Define a transaction modifier.
- Edit the transaction modifier details.
- Apply asynchronous to synchronous transaction modifiers.
- Retain messages at a hub node.

Understanding the Trans Modifiers Page

Access the Relationships — Trans Modifiers page.



Relationships — Trans Modifiers page: Initial tab



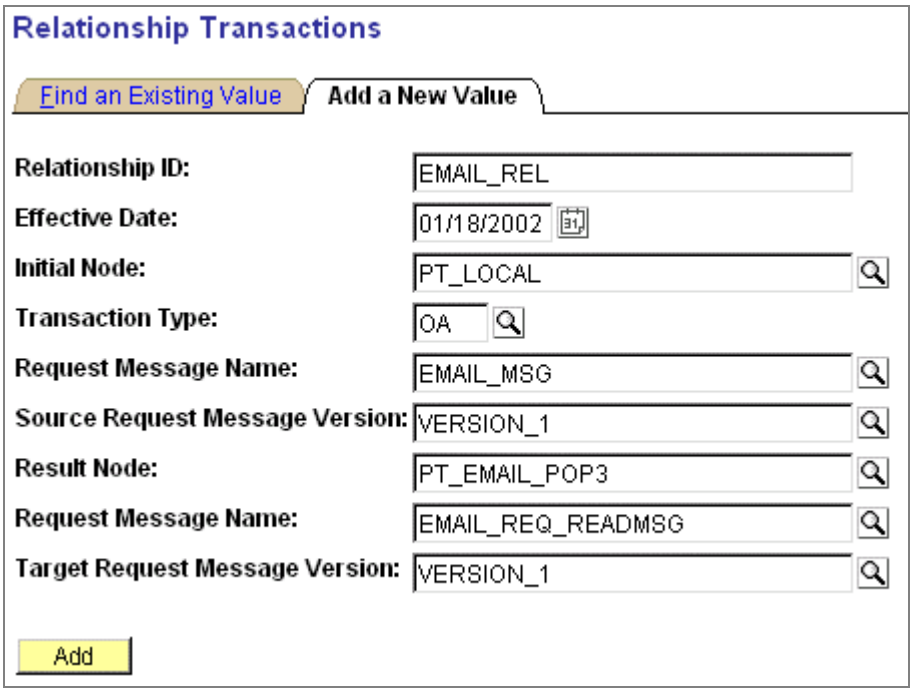
Relationships — Trans Modifiers page: Result tab

Edit Select to modify some properties of the transaction modifier. The Transaction Modifier Detail page appears. See Editing Transaction Modifier Details.

Add Transaction Modifier Click to define a new transaction modifier. The Relationship Transactions page appears. See Defining a Transaction Modifier.

Defining a Transaction Modifier

Use the Relationship Transactions page to specify the basic parameters of a new transaction modifier, including the initial and resulting transactions.



Relationship Transactions page

Relationship ID Enter the identifier for a relationship defined in the current database. The identifier for the current relationship is already entered, but you can select a different one with which to associate this transaction modifier.

Note. If you enter a different relationship ID than the current one, the selected relationship's definition opens when you save the transaction modifier.

Effective Date Select the date this transaction modifier should go into effect. The current date is the default value.

(Initial/Result) Node For both the initial and the resulting transaction, select the node with which each transaction is associated. These must be the two nodes specified for the selected relationship.

Transaction Type For the initial transaction, select from the following types:

Outbound Asynchronous: The default local node sends a request message to the selected node.

Outbound Synchronous: The default local node sends a request message to the selected node, requiring a response.

Inbound Asynchronous: The default local node receives a request message from the selected node.

Inbound Synchronous: The default local node receives a request message from the selected node, requiring a response.

Request Message Name	For both the initial and the resulting transaction, select the name of the request message each transaction should transmit.
(Source/Target) Request Message Version	For both the initial and the resulting transaction, select the version of the request message each transaction should transmit.

Only transactions included in the selected nodes' definitions can be used by this relationship. As you select each parameter for a transaction (node, transaction type, request message name, request message version), the available choices for the remaining parameters are reduced to those that combine with your current choices to define existing transactions. For example, if you select a node with only inbound asynchronous transactions defined, Inbound Asynchronous is the only transaction type available for you to select.

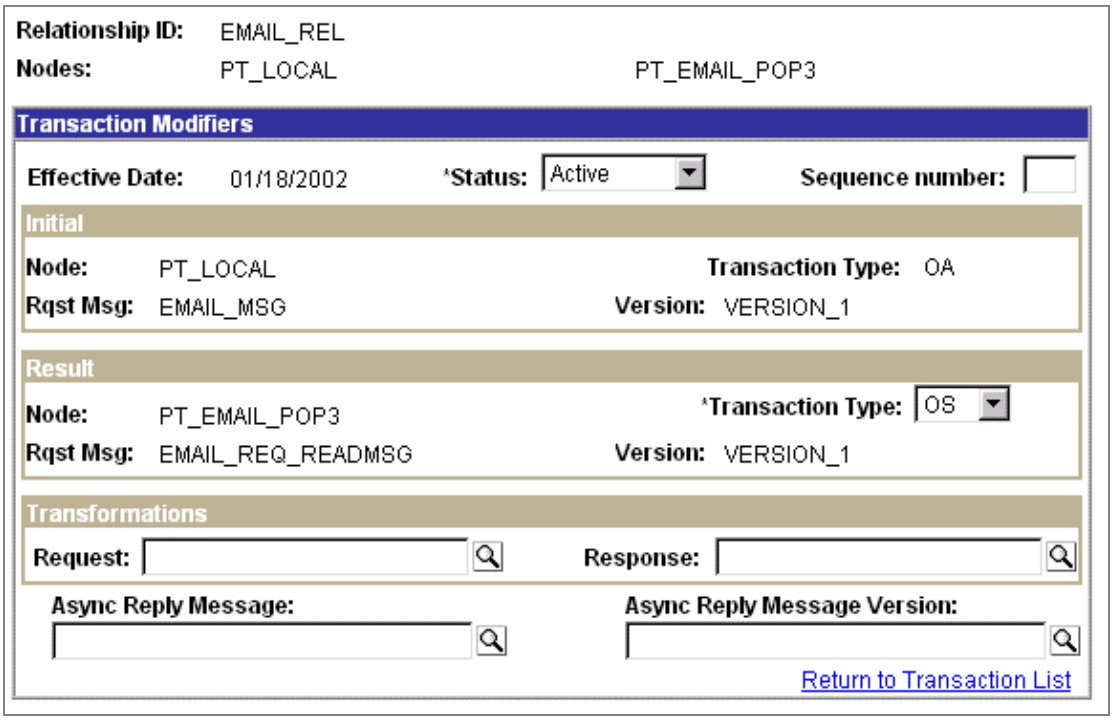
Note. The request messages used by a transaction modifier must be in the same channel.

Click the **Add** button to confirm the parameters you entered for the new transaction modifier. The Transaction Detail page appears.

Editing Transaction Modifier Details

The Transaction Modifier Detail page appears when you select Edit on the Relationships — Transaction Modifiers page, or click the Add button on the Specify Transaction Modifiers – Add a New Value page.

Note. The transaction modifier's basic parameters are read-only, and can be altered only by defining a new transaction modifier with different values. The Async Reply Message and Async Reply Message Version fields appear only if you selected an asynchronous initial transaction and a synchronous resulting transaction.



Transaction Modifier Detail page

Status Select from the following:

Active: When active, this transaction modifier can be applied when the current relationship is invoked. This is the default setting.

Inactive: When inactive, this transaction modifier won't be applied when the current relationship is invoked.

Sequence number (Optional) You can assign sequence numbers to transaction modifiers to indicate the order in which they should be applied. Those without a sequence number are invoked last, in the order they appear on the Trans Modifiers page.

(Result) Transaction Type For the result transaction, select from the following types:

Outbound Asynchronous: The default local node sends a request message to the selected node.

Note. Within a given relationship, you can't select an OutAsync transaction type for two transaction modifiers with the same initial transaction.

Outbound Synchronous: The default local node sends a request message to the selected node, requiring a response.

Inbound Asynchronous: The default local node receives a request message from the selected node.

	Inbound Synchronous: The default local node receives a request message from the selected node, requiring a response.
(Transformations) Request	To apply a transform program to the request message, select the name of the transform program.
(Transformations) Response	(Synchronous only) To apply a transform program to the synchronous response message, select the name of the transform program. **Note.** For synchronous transactions, you may need to apply a transform program to both the request and response messages, although there's no reqirement that they both be applied by the same node.
Async Reply Message	(Asynchronous to synchronous modifier only) Enter the name of the message to be asynchronously returned to the sending node upon receipt of the synchronous response.
Async Reply Message Version	(Asynchronous to synchronous modifier only) Enter the version of the message to be asynchronously returned to the sending node upon receipt of the synchronous response.
Return to Transaction List	Select Return to Transaction List, and the Transaction Modifiers page for the selected relationship appears. **Note.** Be sure to save the transaction modifier before returning to the transaction modifier list. If you don't, your changes — including the creation of a new transaction modifier — will be canceled.

When Integration Broker invokes a transaction whose parameters match those of the initial transaction specified in this transaction modifier, the transaction modifier is applied to produce and invoke the specified resulting transaction. A given transaction modifier is applied only if it and its parent relationship are both active.

Activating Transaction Modifiers

Transaction modifiers are dependent on their parent relationships in the following ways:

- When you deactivate a relationship, all associated transaction modifiers are deactivated.
- You can create new transaction modifiers for inactive transactions, but such transaction modifiers can be saved only with inactive status.

Note. Activating the relationship or both transactions won't automatically reactivate the associated transaction modifiers; they must be individually reactivated.

Applying Asynchronous to Synchronous Modifiers

When you apply an asynchronous to synchronous transaction modifier, the request message is received by the target node in synchronous mode, so that node sends a synchronous response. The only way the source node can accept a response message is in asynchronous mode, so the transaction modifier converts the synchronous response into an asynchronous request, and sends it to the source node as an independent asynchronous transaction in response to the original transaction. You specify this request with the Async Reply Message and Async Reply Message Version fields on the Transaction Modifier Detail page.

Note. The message you specify must have the same structure as the response message sent by the target node, and both must be in the same channel as the initial request message.

You can define and apply the transaction modifier at either the source node or the target node. Because the asynchronous response message is technically unrelated to the initial asynchronous request, you must define appropriate transactions at both the source and target nodes to accommodate it. The following examples describe the transactions required for an exchange in which node **A** sends message **X** to node **B**, and receive message **Y** in response.

The Source Node Applies the Transaction Modifier

In the node A database, an OutAsync to OutSync modifier specifying async response message MSG_Y is defined as part of the NODE_B to NODE_B relationship. To apply the modifier, you must have defined the following transactions:

Database	*Node Definition*	*Transaction Type*	*Message*	*Comment*
A	NODE_B	OutAsync	MSG_X	Source sends async request
A	NODE_B	OutSync	MSG_X	Modifier produces sync request
B	NODE_A	InSync	MSG_X	Target receives sync request
B	NODE_A	OutSync	MSG_Y	Target sends sync response
A	NODE_B	InAsync	MSG_Y	Modifier produces async response

The Target Node Applies the Transaction Modifier

In the node B database, an InAsync to InSync modifier specifying async response message MSG_Y is defined as part of the NODE_A to NODE_A relationship. To apply the modifier, you must have defined the following transactions:

Database	*Node Definition*	*Transaction Type*	*Message*	*Comment*
A	NODE_B	OutAsync	MSG_X	Source sends async request

Database	Node Definition	Transaction Type	Message	Comment
B	NODE_A	InAsync	MSG_X	Target receives async request
B	NODE_A	InSync	MSG_X	Modifier produces sync request
B	NODE_A	OutAsync	MSG_Y	Modifier produces async response
A	NODE_B	InAsync	MSG_Y	Source receives async response

Retaining Messages at a Hub Node

If you're using an inbound to outbound transaction modifier for message routing at a hub node, you may want the hub node to retain (receive and process) the message locally as well. You define an additional transaction modifier to accomplish this. The following example specifies the transactions required for an exchange in which node **A** sends message **X** to hub node **B**, which routes the message to node **C**, and consume the message itself as well.

Applying Basic Routing

In the node B database, an InAsync to OutAsync modifier is defined as part of the NODE_A to NODE_C relationship. To apply the modifier, you must have defined the following transactions:

Database	Node Definition	Transaction Type	Message	Comment
A	NODE_B	OutAsync	MSG_X	Source sends async request
B	NODE_A	InAsync	MSG_X	Hub receives async request
B	NODE_C	OutAsync	MSG_X	Hub sends async request
C	NODE_B	InAsync	MSG_X	Target receives async request

Retaining the Message

For node B to receive and process the same message, the appropriate InAsync transaction is already defined in the node B database, as shown in the table. Additionally, the node B database must contain a NODE_A to NODE_A relationship which includes an InAsync to InAsync modifier. The modifier essentially routes the message back through the original transaction, at which point node B can receive and process it normally.

Note. Integration Broker doesn't support applying multiple transaction modifiers in sequence to a single message. However, you can apply them in parallel, as in this example.

CHAPTER 12

Using the Integration Gateway

This chapter discusses how to:

- Set Integration Gateway properties.
- Work with target and listening connectors.
- Set up gateway certificates.
- Run the Integration Gateway behind a proxy web server.

Setting Integration Gateway Properties

The IntegrationGateway.properties file enables you to specify and set gateway properties to modify gateway behavior, without having to recompile source code.

This section describes all of the properties contained in the IntegrationGateway.properties file. The location of the file depends on the web server that you are using.

- WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Web-inf\integrationGateway.properties

- WebSphere:

c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Web-inf\integrationGateway.properties

Note. After you change this file, you mush refresh the gateway for your changes to take effect.

This section discusses how to:

- Set Integration Gateway version properties.
- Set Integration Gateway class installation properties.
- Set Delivered Connector configuration properties.
- Set logging properties.

- Set proxy web server properties.
- Set Integration Gateway certificate properties.
- Set Java Messaging Service (JMS) configuration properties.

See Also

"Administering Basic Integrations," Refreshing the Gateway Properties

Setting Integration Gateway Version Properties

You must specify the following version setting for the Integration Gateway.

ig.version	Determines the version of the Integration Gateway.

Note. The version number you enter must contain two decimal places, for example: 8.40

Setting Integration Gateway Class Installation Properties

You must specify the following class installation setting for the Integration Gateway.

ig.installdir	Identifies the location of installed Integration Gateway classes. This information is used on introspection calls to determine the location, name and properties of target connectors.

Setting Delivered Connector Configuration Properties

The Delivered Connector Configuration Section of the IntegrationGateway.properties file includes properties you must set for target connectors delivered with PeopleSoft Integration Broker and the default application server. You must also set Jolt connect string properties for each PeopleSoft 8.4 application server node with which you want the Integration Gateway to communicate.

Delivered Connector Settings

In the Delivered Connector Configuration section of the IntegrationGateway.properties file, you set properties for the target connectors delivered with PeopleSoft Integration Broker and that require gateway-level property settings, including:

- HTTP Target Connector
- PeopleSoft Target Connector
- Simple Mail Transfer Protocol (SMTP) Target Connector

- PeopleSoft 8.1 Target Connector

Jolt Connect String Settings

You must set Jolt connect string properties for each PeopleSoft 8.4 application server node with which you want the Integration Gateway to communicate.

For any inbound message to the PeopleSoft Integration Broker, the only information that the Integration Gateway requires is the message name and requesting node. (Node, in this case, is an Extensible Markup Language [XML] term for a tag.)

When a message is sent to the PeopleSoft Integration Broker from a PeopleSoft system, in addition to the requesting node and message name, it always populates the <TO> node as follows:

```
<To>
<DestinationNode>nodename</DestinationNode>
</To>
```

However, 99-percent of the time inbound messages from third-party systems will not contain this information. So in most cases when the PeopleSoft Integration Broker receives messages from a third-party system, the Integration Gateway does not know to which nodes to send the messages.

If there is no <TO> node populated in an inbound message, the Integration Gateway looks for the default Jolt Connect string set in ig.isc.server.URL property.

If there is a <TO> node populated in the inbound message, the Integration Gateway looks for a match with $NODENAME in the ig.isc.$NODENAME.serverURL property, and performs one of the following:

- If it finds a match, then it uses the Jolt Connect string specified.
- If it does not find a match, it looks for the default Jolt Connect string set in ig.isc.server.URL property.
- If there is no default set, then it generates an exception.

Default Target Connector Property

Whenever a message reaches the Integration Gateway and its content does not define a <CONNECTOR> node (which should always be the case when you receive a message from a third-party system) the Integration Gateway automatically uses the connector specified in the ig.connector.ibtargetconnector setting. By default, this property is set to the PeopleSoft Target Connector.

You specify the following properties in the Delivered Connector Properties section of the IntegrationGateway.properties file.

ig.connector.prefix	Identifies the Universal Resource Indicator (URI) prefix added to any connector name. This property instantiates the connector classes on your system. PeopleSoft recommends that you do not change this value.
ig.connector.defaultremoteconnector	Use to send messages to a remote Integration Gateway. The default connector is the HTTP Target Connector (HTTPTARGET). PeopleSoft recommends that you do not change this value.
ig.connector.ibtargetconnector	Use to send messages to PeopleSoft 8.4 systems. This is the connector the Integration Gateway uses to connect to the Integration Engine running on the application server. The default connector is the PeopleSoft Target Connector (PSFTTARGET). PeopleSoft recommends that you do not change this value.
ig.connector.smtptargetconnector.host	Identifies the SMTP host used by the SMTP Target Connector (SMTPTARGET). This property is required only if you are sending messages to an SMTP mail server. If you are doing so, uncomment this property and set the value equal to the SMTP mail server name.

ig.isc.serverURL	Identifies the URL for the default application server.
	
	Use this property if no <TO>nodename</TO> is specified in the message, or if the Node name specified does not match with one of the $NODENAMEs in ig.isc.$NODENAME.serverURL entries. The format is:
	`//<machine name>:<jolt port>`
	
	For an application server setup for failover or with multiple domains, you can specify a list of server URLs in comma delimited format. For example:
	`//<machine name>:<jolt port>,<machine name>:<jolt port>`

ig.isc.userid	Identifies the user ID for the PeopleSoft database on the default application server.

ig.isc.password	Identifies the password for the PeopleSoft database user on the default application server.
ig.isc.toolsRel	Identifies the PeopleTools release version running on the default application server. For example, 8.40.
ig.isc.$NODENAME.serverURL	Identifies the URL (Jolt connect string) that the Gateway Manager uses to connect to a PeopleSoft 8.4 application server destination node. The format is: `//<machine name>:<jolt port>` You must also change $NODENAME to the EXACT name of the destination node. Note: The pssappsvr.cfg file on the application server contains Jolt port number information.
ig.isc.$NODENAME.userid	Identifies the database user ID for the PeopleSoft 8.4 destination node. You must also change $NODENAME to the EXACT name of the destination node.
ig.isc.$NODENAME.password	Identifies the database password for the PeopleSoft 8.4 destination node. You must also change $NODENAME to the EXACT name of the destination node.
ig.isc.$NODENAME.toolsRel	Identifies the PeopleTools release version for the destination node. You must also change $NODENAME to the EXACT name of the destination node.
ig.connector.amtargetconnector.url	Identifies the URL used by the PeopleSoft 8.1 Target Connector to send messages to the default PeopleSoft 8.1 Application Messaging gateway. You can override this default at runtime by specifying a different URL in a Node Definition component. You must define this property only for integrations with PeopleSoft 8.1 systems.

See Also

Using the HTTP Target Connector

Using the PeopleSoft Target Connector

Using the SMTP Target Connector

Using the PeopleSoft 8.1 Target Connector

Setting Logging Properties

The following are general, error, and message logging properties for the Integration Gateway.

General Properties

ig.log.level

Numeric value that determines the level of gateway logging and exception handling.

Valid values:

–100: Suppress message logging.

–1: Log language exceptions only.

1: Log language and standard exceptions.

2: Log all errors and warnings.

3: (Default) Log errors, warnings, and important information.

4: Log errors, warnings, and important and standard information.

5: Log errors, warnings, and important, standard, and low-importance information.

The default value is 5.

ig.log.backgroundImage

Background image used for the error and message log documents. The default image name and location depends on the web server installed. The default file location depends on the web server installed.

WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\PSbackground.jpg

WebSphere:

c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\PSbackground.jpg

Message Logging Properties

ig.messageLog.filename

Identifies the message log location. The default location depends on the web server that is installed.

WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\msgLog.html

	WebSphere: c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\msgLog.html
ig.messageLog.maxSize	Identifies the maximum size of the message log, in kilobytes (KB). The default is 10,000 KB, or 10 megabytes (MB). When the limit is reached, the log is archived, and a timestamp is appended to the file name.
ig.messageLog.maxNbBackupFiles	Identifies the number of archived files to keep on disk. The default value is *5*. The value *0* keeps all backed up files.

Error Logging Properties

ig.errorLog.filename	Identifies the error log location. The default location depends on the web server installed. WebLogic: c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\errorLog.html WebSphere: c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\errorLog.html
ig.errorLog.maxSize	Identifies the maximum size of the message log, in KB. The default is 10,000 KB, or 10 MB. When the limit is reached, the log is archived, and a timestamp is appended to the file name.
ig.errorLog.maxNbBackupFiles	Identifies the number of message archived files to keep on disk. The default value is *5*. The value *0* keeps all backed up files.

See Also

Integration Gateway Message and Error Logging

Setting Proxy Web Server Properties

You must specify the properties if you are running the Integration Gateway behind a proxy web server.

ig.proxyHost	Identifies the domain name of the proxy web server. For example: proxy.peoplesoft.com.
ig.proxyPort	Identifies the port number of the proxy web server. For example, 80.

See Also

Running the Integration Gateway Behind a Proxy Server

Setting Integration Gateway Certificates Properties

You must set the following properties to set up gateway certificates.

One of the properties you must specify is an encrypted certificate password. PeopleSoft provides a utility for encrypting passwords. See Setting Up Gateway Certificates

ig.certificateAlias	Identifies the certificate alias.
ig.certificatePasswd	Identifies the certificate password. The certificate password must be encrypted.
SecureFileKeystorePath	Identifies the directory path to the keystore file. The default path for the keystore depends on the web server that is installed. WebLogic: • c:/bea/wlserver6.1/config/peoplesoft/keystore/pskey WebSphere: • c: /Apps/WebSphere/AppServer/installedApps/peoplesoft/keystore/pskey
secureFileKeystorePasswd	Identifies the keystore password. This password is not encrypted.

Setting JMS Configuration Properties

This section describes properties to set if you are using a Java Messaging Service (JMS) Target Connector or a JMS Listening Connector.

For the JMS Listening Connector you can configure multiple queues and topics. To configure multiple queues, use the convention, *ig.queue1*, *ig.queue2*, *ig.queue3*, and so on. To configure multiple topics, use the convention *ig.topic1*, *ig.topic2*, *ig.topic3*, and so forth.

When you set the JMSProvider property, the provider you enter must match the provider in the JNDIFactory Classname exactly. You must set this property for both the JMS Listening Connector and the JMS Target Connector. This property is case-sensitive.

PeopleSoft currently supports JNDI only for File System Context [fscontext].

JNDIFactory Classnames

ig.jms.JMSProvider.JNDIFactory .Weblogic	Identifies the JNDIFactory class name for a WebLogic web server. Uncomment this property if you are using a WebLogic web server.

	The default value is: • weblogic.jndi.WLInitialContextFactory
ig.jms.JMSProvider.JNDIFactory.iPlanet	Identifies the JNDIFactory class name for an iPlanet web server. Uncomment this property if you are using an iPlanet web server. The default value is: • com.sun.jndi.fscontext.RefFSContextFactory
ig.jms.JMSProvider.JNDIFactory.MQSeries	Identifies the JNDIFactory class for an MQSeries server. Uncomment this property if you are using an MQSeries server. The default value is: • com.sun.jndi.fscontext.RefFSContextFactory

JMS Queue Listener Properties

ig.jms.Queues	Identifies the number of Queue Listeners to instantiate.
ig.jms.Queue1	Identifies the queue name.
ig.jms.Queue1.Provider	Identifies the queue provider name.
ig.jms.Queue1.JMSFactory	Identifies the JMSFactory name that is bound to JNDI for the queue.
ig.jms.Queue1.MessageSelector	Identifies the optional message filter.
ig.jms.Queue1.URL	Identifies the JMS provider's URL to JNDI
ig.jms.Queue1.User	Identifies the JMS queue user name.
ig.jms.Queue1.Password	Identifies the JMS queue password.

JMS Topic Subscriber Properties

ig.jms.Topics	Identifies the number of Topic Subscribers to instantiate.
ig. jms.Topic1	Identifies the Topic name.
ig. jms.Topic1.Provider	Identifies the Topic provider name.
ig. jms.Topic1.JMSFactory	Identifies the JMSFactory name that is bound to JNDI for the topic.
ig. jms.Topic1.MessageSelector	Identifies the message filter.
ig. jms.Topic1.URL	Identifies the JMS provider's URL to JNDI.
ig. jms.Topic1.User	Identifies the JMS Topic user name.
ig. jms.Topic1.Password	Identifies the JMS Topic password.

Error-Topic configuration

ig.jms.ErrorTopic	Identifies the name of Topic to which to publish error messages.
ig.jms.ErrorTopic-Provider	Identifies the name of the JMS Provider.
ig.jms.ErrorTopic-User	Identifies the JMS Error Topic user name.
ig.jms.ErrorTopic-Password	Identifies the JMS Error Topic password.
ig.jms.ErrorTopic-JMSFactory	Identifies the JNDI Factory name.
ig.jms.ErrorTopic-Url	Identifies the JMS provider's URL to JNDI.

See Also

Using the JMS Target Connector

Using the JMS Listening Connector

Understanding Message Compression and Encoding

The Integration Engine always compresses and base64 encodes asynchronous messages destined for the Integration Gateway. By default the message is decoded and deflated before it leaves the gateway and reaches its recipient.

The only exception to this rule is if a message is destined for a remote gateway. In this case the message remains compressed and encoded. If a third party requires messages to remain compressed and encoded, you need to specify so in the Node Definition component by setting the SendUncompressed connector property to *No*. However, if nonrepudiation is in effect, the SendUncompressed connector property will not be taken into consideration and messages will always be sent compressed and base64 encoded.

Synchronous messages are handled in the same manner. In addition, you can specify when compression and encoding is to occur for synchronous messages from the application server configuration file, psappsrv.cfg, located in the PS_HOME/appserv/DomainName directory. The possible settings are:

- Compress and encode when the message is larger than a specified size.
- Always compress.
- Never compress.

For synchronous messages, if nonrepudiation is in effect, the SendUncompressed connector property is not applicable because these types of messages are decompressed and decoded by default.

Working with Target Connectors

This section discusses:

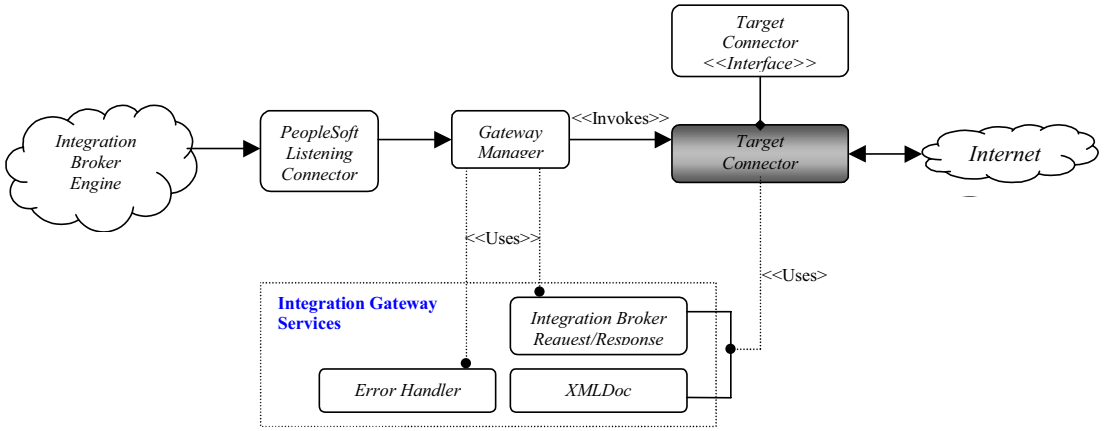
- How messages flow through target connectors on the Integration Gateway.
- Message encoding and compression.
- Types of target connector properties.
- HTTP Target Connector and its properties.
- PeopleSoft Target Connector and its properties.
- PeopleSoft 8.1 Target Connector and its properties.
- FTP Target Connector and its properties.
- SMTP Target Connector and its properties.
- JMS Target Connector and its properties.
- POP3 Target Connector and its properties.
- Simple File Target Connector and its properties.

Understanding Target Connectors

This section discusses:

- The flow of messages through target connectors on the Integration Gateway.
- PeopleSoft-delivered target connectors.

Target connectors generate message requests, send them to integration participants, wait for responses from participants and deliver the responses back to the Gateway Manager, as shown in the diagram.



Message flow through a target connector on the Integration Gateway.

The Integration Gateway invokes target connectors dynamically via the Gateway Manager. Target connectors adhere to a standard structure by implementing the Target Connector

Interface provided by the Integration Gateway. By implementing this interface, target connectors can take advantage of all Gateway Manager services.

PeopleSoft-Delivered Target Connectors

PeopleSoft delivers eight target connectors with the PeopleSoft Integration Broker that enable you to communicate with integration participants using a number of communication formats. The target connectors that PeopleSoft delivers with the PeopleSoft Integration Broker are:

- HTTP Target Connector.
- PeopleSoft Target Connector.
- PeopleSoft 8.1 Target Connector.
- FTP Target Connector.
- SMTP Target Connector.
- JMS Target Connector.
- POP3 Target Connector.
- Simple File Target Connector.

Understanding Target Connector Properties

Most of the delivered target connectors have required and optional properties. You set some of these properties at the node level, and others at the Integration Gateway level.

This section discusses:

- Target connector properties you set at the node level.
- Target connector properties you set at the gateway level.

Node-Level Connector Properties

Set node-level connector properties in the Gateways component. For each connector that is described in this section, node-level properties are described in detail, where applicable.

Each property has a property ID (PROPID) a property name (PROPNAME), a default flag and a value.

Node-level properties are sent along with the message to the Integration Gateway to provide information to the intended Target Connector about how to process the message.

The three target connectors that communicate over HTTP are the:

- HTTP Target Connector
- PeopleSoft Target Connector

- PeopleSoft 8.1 Target Connector

For each of these connectors, you can specify only one primary URL (PRIMARYURL) per node. The **primary URL** is the URL of the external system that will handle the request.

However, you may specify more than one backup URL (BACKUPURL). Upon the failure of a transaction to the primary URL, the message is sent to any **backup URLs**, one at a time. When a transaction that is sent to a URL succeeds, the other URLs are not used. If all URLs fail, the appropriate action and message is relayed to the calling module, and the message and node/URL failure is noted in the database or in the Integration Broker Monitor.

Note. If the Property ID is *HEADER* then the target connector retrieves the information from a getHeader method call on the ConnectorInfo object, which resides on the IBRequest object. All other properties can be retrieved from a getFieldValue method call on the ConnectorInfo.

Gateway-Level Connector Properties

Set gateway-level connector properties in the IntegrationGateway.properties file. This section describes gateway-level properties for each connector in detail, where applicable.

See Also

Setting Integration Gateway Properties

Using the HTTP Target Connector

The **HTTP Target Connector** enables you to send messages to and receive replies from non-PeopleSoft systems using the HTTP protocol. The HTTP Target Connector uses Secured Socket Layer (SSL) for all basic security services, including client-side authentication.

The HTTP Target Connector also supports the Simple Object Access Protocol (SOAP) XML format.

The connectorID for the HTTP Target Connector is HTTPTARGET.

Node-Level Connector Properties

The HTTP Target Connector features properties that adhere to the Hypertext Transfer Protocol -- HTTP/1.1, as well as custom properties.

This section describes required and custom properties for the HTTP Target Connector. The World Wide Web Consortium (W3C) web site provides descriptive information about the properties that adhere to the Hypertext Transfer Protocol. See http://www.w3.org/Protocols/rfc2616/rfc2616.html

Use the Gateways component to view and set all node-level properties for this connector.

Property ID	Property Name	Description
HEADER	SendUncompressed	Enables you to send messages decompressed. The valid values are: • *Y*: (default) Send messages compressed. • *N:* Send messages decompressed.
HEADER	SOAPAction	(Optional) Enables third-party systems (for example, UDDI sites) to receive SOAP transactions over HTTP. The default value is *" "*.
HEADER	Proxy-Authenticate	(Optional) Identifies the user ID and password for proxy authentication. After you set this information, the Integration Gateway encodes the user ID and password, adds the required formatting and sends it. The format is: `userid:pwd`
HTTPPROPERTY	Method	Method performed on the resource identified by the Request-URI. The valid values are: • *POST* (default) • *GET* Values are case-sensitive.
PRIMARYURL	URL	Identifies the URL of the external system that handles the request.
BACKUPURL	URL	(Optional) Identifies the backup URL of the external system that handles the request.

Gateway-Level Connector Properties

As an option, the HTTP Target Connector supports routing through proxy servers. To enable this capability, you must set two properties in the IntegrationGateway.properties file.

```
ig.proxyHost=machine_name
ig.proxyPort=proxy_port_number
```

The HTTP Target Connector reads these two properties and calls the **setProxy** function. In an actual outbound call, the request is redirected to the proxy server and the proxy server forwards the request to the destination URL.

Proxy authentication is handled using the HEADER – proxy_authentication in the connector Properties entry in the node definition.

See Also

Running the Integration Gateway Behind a Proxy Server

Using the PeopleSoft Target Connector

The **PeopleSoft Target Connector** initiates conversation with the Integration Engine through a Jolt call. You can configure your Integration Gateway to dispatch messages to different Integration Engines based on the destination node that is specified on the incoming message. Set the destination node properties in the Application Server Settings section of the IntegrationGateway.properties file.

The connectorID for the PeopleSoft Target Connector is PSFTTARGET.

Node-Level Connector Properties

There are no node-level connector properties for the PeopleSoft Target Connector.

Gateway-Level Connector Properties

The destination node (known node) settings in the Application Server Settings section of the IntegrationGateway.properties file determine the Jolt connect string that is used to reach the Integration Engine.

Using the PeopleSoft 8.1 Target Connector

The **PeopleSoft 8.1 Target Connector** enables you to communicate between PeopleSoft 8.4 and PeopleSoft 8.1 systems. Messages from the PeopleSoft 8.4 system reach the PeopleSoft 8.1 system through the Application Messaging Gateway on the PeopleSoft 8.1 system. The communication takes place over HTTP(S). You can use client-side digital signatures and certificates in the PeopleSoft 8.4 system. To do so, you set the Authentication option in the Node Definition component to *Certificate*.

The PeopleSoft 8.1 Target Connector uses the HTTP Target Connector to manage the HTTP communication with the PeopleSoft 8.1 Application Messaging Gateway. The PeopleSoft 8.1 Target Connector focuses on messaging semantics, instead of communication details, and constructs an Application Messaging XML document and sends it to the PeopleSoft 8.1 Application Messaging Gateway.

You specify the destination PeopleSoft 8.1 Application Messaging Gateway in the Gateway component. If none is specified, the connector uses the default value specified in the IntegrationGateway.properties file.

The PeopleSoft 8.1 Target Connector detects the status of returned responses by the value in the ReturnCode field in the XML response.

The connectorID for the PeopleSoft 8.1 Target Connector is PSFT81TARGET.

Node-Level and Integration Gateway-Level Connector Properties

Property ID	*Property Name*	*Description*
PSFT81Target	URL	Identifies the HTTP address for the target Application Messaging Gateway.

The PeopleSoft 8.1 Target Connector has only one property or parameter, the destination URL. The destination URL refers to the PeopleSoft 8.1 Application Messaging Gateway URL.

You can set the destination URL on a case-by-case basis in the Node Definitions component, or use the ig.connector.amtargetconnector.url property in the IntegrationGateway.properties file to set a default URL to use for all transactions.

Using the FTP Target Connector

The **FTP Target Connector** enables you to use the File Transfer Protocol (FTP) to send messages from Integration Broker to FTP servers.

Note. FTP does not support encryption of any sort. PeopleSoft recommends that you not use this connector to transmit sensitive data. The userid/password configured for FTP does not secure message transmission.

The FTP Target Connector allows you to place (PUT) messages or files from the Integration Gateway onto a remote FTP server.

Outbound messages through the FTP Target Connector are UTF-8 encoded.

Required Java Archive (JAR) Files

For the FTP Target Connector to function properly, the FTPProtocol.jar file from IBM must be in the classpath of the Web server running the Integration Gateway.

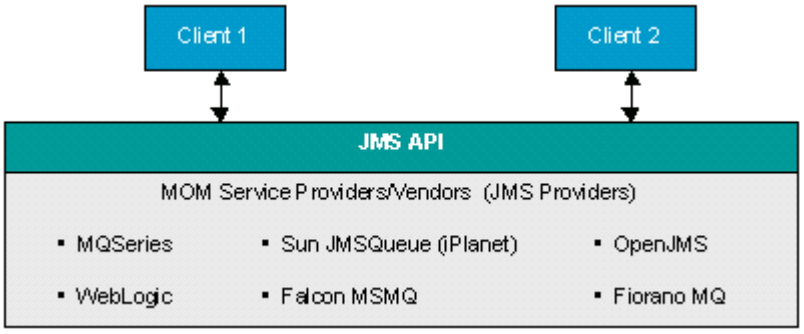
Node-Level Connector Properties

Property ID	*Property Name*	*Description*
FTP TARGET	Hostname	Indicates the IP or name of the FTP server to which to connect.
FTP TARGET	Password	Indicates the password for the login to the FTP server.
FTP TARGET	Username	Indicates the FTP server login ID.
HEADER	SendUncompressed	Enables you to send messages decompressed. The valid values are: • *Y* (Default) • *N*
FTP TARGET	Directory	(Optional) Indicates the remote directory into which the file is placed. If not specified, the default directory of the FTP server on the remote site is used.

Property ID	*Property Name*	*Description*
FTP TARGET	Filename	(Optional) Indicates the name of the file saved on the recipient's FTP server. By default, the filename is a concatenation of the following: • OriginatingNode • OriginatingUser • MessageName • OriginatingTimeStamp

Using the JMS Target Connector

JMS is an API for accessing message systems. JMS provides a standard Java-based interface to the message services of Message-oriented middleware (MOM) providers. The **JMS Target Connector** is an adapter to JMS providers, and can be used with MOM and JMS providers, such as MQSeries, WebLogic, iPlanet and others.



Message flow through the JMS API

The primary features of JMS are.

- Connection factories that are used to create connections to a specific JMS provider.
- Separate publish, subscribe and point-to-point messaging domains.

  These are defined by separate interfaces so that a provider does not have to support both.
- Topics that are used for publish and subscribe, and Queues that are used for point-to-point messaging.

When multiple applications must receive the same message, publish and subscribe messaging is used. In **publish and subscribe messaging** all of the subscribers subscribe to a **Topic** and all of the publishers publish messages to a Topic. The messaging system distributes messages from the publisher to the subscriber. This domain is mainly used for asynchronous messaging.

When one application must send a message to another application, **point-to-point messaging** is used. This domain is used mainly for synchronous messaging. There are two basic types of point-to-point messaging systems. One involves a client that directly sends a message to

another client, and the another more commonly used implementation is based on the concept of a message Queue.

The JMS Target Connector either publishes messages to a Topic or inserts a message into the Queue, based on the properties of the node. JMS Target connector uses properties you set at the node level in the Node Definition component. You set other JMS Target Connector properties in the IntegrationGateway.properties file and in the JMS Message header.

Asynchronous and Synchronous Communication

The JMS Target Connector provides both synchronous and asynchronous modes of communication. When the node level property ReplyTo is set to *False*, communication is asynchronous, and when it is set to *True*, communication is synchronous.

For asynchronous communication, the JMS Target Connector publishes messages to MOM or drops messages into queue and commits the session, and does not wait for a reply from the destination system. For synchronous communication, after the connector publishes messages or drops them into queue, it waits on temporary topic/queue for a reply.

For synchronous communication, the exchanges involve only the publisher and a single subscriber. When the reply is received it checks for the JMS Correlation ID, if it matches to the correlation ID of the message sent, then it processes the reply. So the when an external system receives messages, it must set the correlation ID along with its reply. The connector also sets reference to the Temporary Queue or Topic on which it wants the reply.

When sending messages either synchronously or asynchronously, the connector sets different string properties in the JMS message header. The properties are used as meta data about the message. JMS message header properties are described later in this section.

Node-Level Connector Properties

This section discusses the properties that you set for the JMS Target Connector in the Node Definition component. You must set either a JMSQueue or JMSTopic. If both are set or are missing the PeopleSoft Integration Broker generates an InvalidMessageException.

Note. You must register JMS-administered objects, such as topics, queues and connection factories that you include as connector properties. The administration manuals of specific providers should provide you with instructions on how to register the topics.

JMS Message types can be *Text*, *Map Message*, *Stream* or *Object*. However, Peoplesoft provides only text messages. If you need to use other message types, you can write a class that implements the com.peoplesoft.pt.integrationgateway.common.jms.ProcessJMSMessage interface, and set the class name as a value for JMSMessageTypeClass.

The provider name that you specify for the JMSProvider in the node definition must match the JMSProvider.JNDIFactory property that you specify in the IntegrationGateway.properties file.

Property ID	*Property Name*	*Description*
HEADER	SendUncompressed	Enables you to send messages decompressed. Values are: • *Y*: (Default) Send the message decompressed. • *N*: Send the message compressed.
JMSTARGET	JMSAcknowledgement	Identifies the acknowledgement type. Values are: • *Auto_Acknowledge* (Default) • *Client_Acknowledge*
JMSTARGET	JMSDeliveryMode	Identifies the durable or non-durable delivery. Values are: • *Persistent* • *Non-persistent* (Default)
JMSTARGET	JMSFactory	Identifies the factory name. The default value is *QueueConnectionFactory.*
JMSTARGET	JMSMessageTimeToLive	Identifies the time in seconds.
JMSTARGET	JMSMessageType	Identifies the type of message to send. The valid values are: • *Text* (Default) • *MapMessage* • *Stream* • *Object*
JMSTARGET	JMSPassword	Identifies the password to access the connection.
JMSTARGET	JMS Priority	Identifies the message priority for delivery. Values range from 0 to 9, with 9 indicating the highest priority. The default is *0*.
JMSTARGET	JMSProvider	Identifies the JMS provider's name. Values are: • *MQSeries* (Default) • *Weblogic* • *iPlanet*
JMSTARGET	JMSReplyTo	Set this property to *True* to receive a reply from the external system. Values are: • *True* • *False* (Default)
JMSTARGET	JMSUrl	Identifies the URL

Property ID	Property Name	Description
JMSTARGET	JMSUserName	Identifies the username to establish a connection to the JSM
JMSTARGET	JMSTopic	(Optional) Identifies the Topic name, if using a Topic. You must either use a Topic or a Queue.
JMSTARGET	JMSQueue	(Optional) Identifies the Queue name, if using a Queue. You must either use a Topic or a Queue.
JMSTARGET	JMSMessageTypeClass	(Optional) Identifies the implementation class of ProcessJMSMessage. You must set this property when the JMSMessageType is anything other than *Text.*

Gateway-Level Connector Properties

To use the JMS Target Connector, you must set several properties in the JMS Configuration section of the IntegrationGateway.properties file.

You can use multiple JMS Providers at a time with different nodes, provided that you set required JMS properties for each node component in the IntegrationGateway.properties file. For example:

NODE_1 publishes to MQSeries and NODE_2 publishes to WebLogic

The JMS Target connector supports only Java Naming and Directory Interface (JNDI) File context for the look up of connection factories, topics and queue names. (Lightweight Directory Access Protocol (LDAP) is not supported.) It uses the JMSConnectionFactory property value to look for a connection factory. You must set either the JMSQueue or JMSTopic property; the value of the property you select is used for looking up the Topic or Queue.

In addition, the ig.jms.JMSProvider.JNDIFactory property specifies which JNDI service provider is in use. File system service providers are available for iPlanet, MQSeries, and Weblogic. In the IntegrationGateway.properties file, the following settings are provided and are commented out. You must uncomment the line that contains the service provider you are using.

```
#ig.jms.JMSProvider.JNDIFactory.Weblogic=weblogic.jndi.WLInitialContextFactory

#ig.jms.JMSProvider.JNDIFactory.iPlanet=com.sun.jndi.fscontext.RefFSContextFac
tory

#ig.jms.JMSProvider.JNDIFactory.MQSeries=com.sun.jndi.fscontext.RefFSContextFa
ctory
```

You can also specify a service provider, other than those listed. For example, if you are using MSMQ, enter the following value for the property:

```
ig.jms.JMSProvider.JNDIFactory.MSMQ=com.sun.jndi.fscontext.RefFSContextFactory
```

Note. The provider name that you specify for the JMSProvider.JNDIFactory property must match the provider name of the JMSProvider that you set in the node definition.

Message Header Properties

When a JMS message is published the following string properties are set in the JMS message header. These properties are Peoplesoft-specific and are sent by the JMS Target Connector as credentials for the benefit of the message recipient.

MessageName	Indicates the name of the message.
RequestingNode	Indicates the requesting node name.
FinalDestinationNode	Indicates the final destination node(s). If there are no values, set to *Null*.
DestinationNode	Indicates the destination node name(s). Use a comma-separated string if there are multiple values. If there are no values, set to ″ ″ (empty string).
Password	Indicates the password for the recipient system.
ReplyTo	Indicates whether a reply is required. Values are: • *True*: Specifies that a reply is required. Sets up synchronous publishing. • *False*: Specifies that a reply is not required. Sets up asynchronous publishing.

You set the following properties in the JMS Message header if the message is compressed or encoded. To compress the message, set the property sendUncompressed to *N* at the node component level.

encoding	Base64(deflate). Peoplesoft-specific encoding
length	Indicates the length, in bytes, of the message in its decompressed format
encodedlength	Indicates the length, in bytes, of the message in its compressed format

Required Java Archive (JAR) Files

To use the JMS Target Connector, you must add specific Java Archive (JAR) files to the java CLASSPATH. The JAR files you add to the CLASSPATH depend on the web server that you are using.

WebLogic	Weblogic.jar
iPlanet	jms.jar, ,jmq.jar, fscontext.jar, jndi.jar
MQSeries	jms.jar, ,jndi.jar, fscontext.jar. com.ibm.mqjms.jar

Additional Setup Steps

Before using the JMS Target Connector, verify that:

1. The JMS messaging system is running.
2. All the JMS connection factories, Topics and Queues are registered for JNDI look up.
3. A username and a password are created in the JMS system for use as values for the properties JMSUserName and JMSPassword.

JMS Target Connector Errors and Exceptions

The JMS Target Connector may generate the following exceptions..

InvalidMessageException	Generated when any of the node level or connector parameters are not set properly. Examples are: • Both Queue and Topic are specified. • Neither Queue nor Topic is specified. • A JMSSecurityException is generated. (Verify username and password are correct.) • When a naming exception occurs.
ExternalApplicationException	Generated when: • The correlation ID does not match when the ReplyTo property is set *True*. • The message could not put into Queue or unable to publish to a topic.
GeneralFrameWorkException	Generated when a NamingException occurs.

Using the SMTP Target Connector

The **SMTP Target Connector** enables you to send messages by e-mail through the SMTP. The SMTP Target Connector supports text/plain and text/html content types.

The connector supports the following fields: To:, cc:, bcc: and Subject:. You may send data of any format in the body of the e-mail.

You may include only one email address per type of address in the header. For instance, you may include only one addressee as a destination (DestEmailAddress).

Node-Level Connector Properties

Property ID	*Property Name*	*Description*
HEADER	Content-Type	(Optional) Identifies the type of text content that makes up the e-mail body. Values are: • *Text/plain* • *Text/html*
HEADER	SendUncompressed	Enables you to send messages decompressed. Values are: • *Y* (Default) • *N*
SMTPTARGET	BCC	(Optional) Identifies the e-mail address of the party to which you are sending blind copies of messages. Only one e-mail address is allowed per address type.
SMTPTARGET	CC	(Optional) Identifies the e-mail address of the party to which you are sending messages. Only one e-mail address is currently allowed per address type.
SMTPTARGET	DestEmailAddress	Identifies the e-mail address to which you are sending messages.
SMTPTARGET	SourceEmailAddress	Identifies the e-mail address from which you are sending messages.

Using the POP3 Target Connector

You can use the **POP3 Target Connector** under two different situations. The first is where a single-user has an account, and the second is where an application has an account.

The POP3 Target Connector supports attachments. Before using the attachments feature, you should make sure that your web server and application server have sufficient memory to handle large attachments.

PeopleSoft provides a default node for use with the POP3 Target Connector, called PT_EMAIL_POP3. More information about this node, including messages and transactions defined for it, is provided later in this section.

POP3 Services

When you use the POP3 Target Connector, you can execute eight services.

When you select and work with the Read Message or Delete Message services, you must specify a UIDL value. **UIDL** is a system used by POP3 servers to uniquely identify a mail message. The UIDL is a fixed String (CHAR, 70) which is unique to the message. The UIDL of a message never changes and will never be reused, even when the message has been deleted from the user's mailbox.

ReadAll	Gets all available messages, including information from the following mail fields and mail message sections: From:, Reply To:, To: address collection, Cc: address collection, Subject:, mail body and UIDL.
ReadAndDelete	Gets all available messages and deletes them from the server, including information from the following mail fields and mail message sections: From:, Reply To:, To: address collection, Cc: address collection, Subject:, mail body and UIDL.
MessageCount	Returns the number of messages in the POP3 server.
ReadMessage	Returns a message for a given UIDL. The response includes information from the following mail fields and mail sections: From: Reply To:, To: address collection, Cc: address collection, Subject, mail body and UIDL.
DeleteMessage	Deletes a message for a given UIDL.
ReadHeaders	Returns information contained in all available message headers, including information from the following mail fields and mail sections: From:, Subject:, UIDL* and received date for the mail message.** * String *NOUIDL* is returned for the UIDL part of the message if the POP3 server does not support UIDL. ** Received date is actually the send date of the mail, and may not be available on all POP3 servers.
ReadHeadersWithAttach	Returns information contained in all available message headers, including information from the following mail fields and mail sections: From:, Subject:, UIDL,* the received date for the mail message, ** and the message attachment list. * String *NOUIDL* is returned for the UIDL part of the message if the POP3 server does not support UIDL. ** Received date is actually the send date of the mail, and may not be available on all POP3 servers.
ReadMessagesWithAttach	Retrieves a specific message with attachments, including information from the following mail fields and mail sections: From:, Reply To: To: address collection, Cc: address collection, Subject:, mail body, UIDL, and message attachment list. It also returns a child record that contains the following information: • **Content Type**: If the message is a multi-part message, this is the content type for this part.

- **File Name**: This is the attachment file name, this is set to content-ID it the file name is not available.
- **File Data**: This is the file data available in the attachment.
- **Message Body**: If a text or html mail message, it includes the message body.

About POP3 Target Connector Properties

For single-user accounts, you set values for server, port, account, user, password and method name in a request message that contains the EMAIL_RES_WRK work record. See Using the PT_EMAIL_POP3 node later in this section.

For application accounts you set connector properties at the node level in the Node Definitions component. You must create separate transactions for the POP3 node for each service, for example ReadAll, ReadAndDelete, MessageCount, and so forth.

Node-Level Connector Properties

There are several required properties you must set for the POP3 Target Connector at the node level. In addition, there are several optional properties you can set as well.

MethodName values are not case-sensitive.

The POP3 Target Connector returns an empty message in addition to the actual message responses for the following MethodName values: ReadAll, ReadDeleteAll, ReadHeaders, ReadHeadersWithAttach and ReadMessage services.

Note. If you are using the POP3 Target Connector to poll for Blackberry Email Responses using the Blackberry Responses Application Engine program WL_RIMRESP, then the POP3 connector property <User> must match the Application Server SMTP setting called SMTPBlackberryReplyTo in the psappsrv.cfg file. If these do not match, the Blackberry Responses Application Engine program WL_RIMRESP will not function properly and it will not retrieve any Blackberry Email Responses. See *PeopleTools 8.4 PeopleBook: PeopleSoft Workflow*, “Designing Blackberry Email Responses”

Property ID	Property Name	Description
POP3TARGET	Count	Indicates the number of messages or headers to fetch. The value *0* indicates to get all messages/headers. The default is *0*. This property applies to the following methods: • ReadAll • ReadHeaders • ReadHeaderswithAttach • ReadAndDelete
POP3TARGET	MethodName	Indicates the POP3 Service to execute. Values are: • *DeleteMessage* • *MessageCount.* (Default) • *ReadAll.* • *ReadAndDelete.* • *ReadHeaders* • *ReadHeadersWithAttach* • *ReadMessage* • *ReadMessageWithAttach*
POP3TARGET	Password	Indicates the password for the mailbox user
POP3TARGET	Port	Indicates the port that the POP3 server uses for the connection. You must specify this value if you are using the user-account approach to pass connector properties through the EMAIL_RES_WRK record. The default is value is *110*.
POP3TARGET	Server	Indicates the POP3 server name.
POP3TARGET	User	Indicates the mailbox user name to use for the POP3 connection.

Passing Node-Level Connector Properties

Most POP3 requests are performed using the **SyncRequest ( )** method.

```
SyncRequest([NodeName])
```

There are two ways to pass node-level properties to the POP3 Target Connector.

- Pass the values defined for the node as defined in the Node Definition.
- Override all the values defined in the Node Definition at runtime. In this case you pass property values via method field records.

In either situation you use SyncRequest ( ) to pass the property values. The following code examples are provided at the end of this section:

- Passing defined node-level connector properties.
- Overriding defined node-level connector properties.

Gateway-Level Connector Properties

There are no POP3 Target Connector properties that you need to set at the Integration Gateway level.

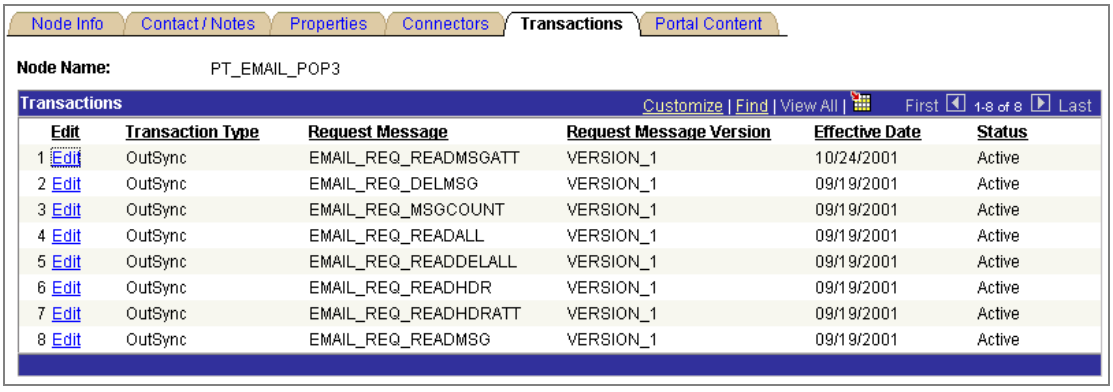
Setting up the POP3 Target Connector Default Node

The POP3 Target Connector default node, PT_EMAIL_POP3, has transactions for each method name value (or service) pre-defined.

To set up the PT_EMAIL_POP3, you must edit each transaction and specify the POP3 server name, user name and password.

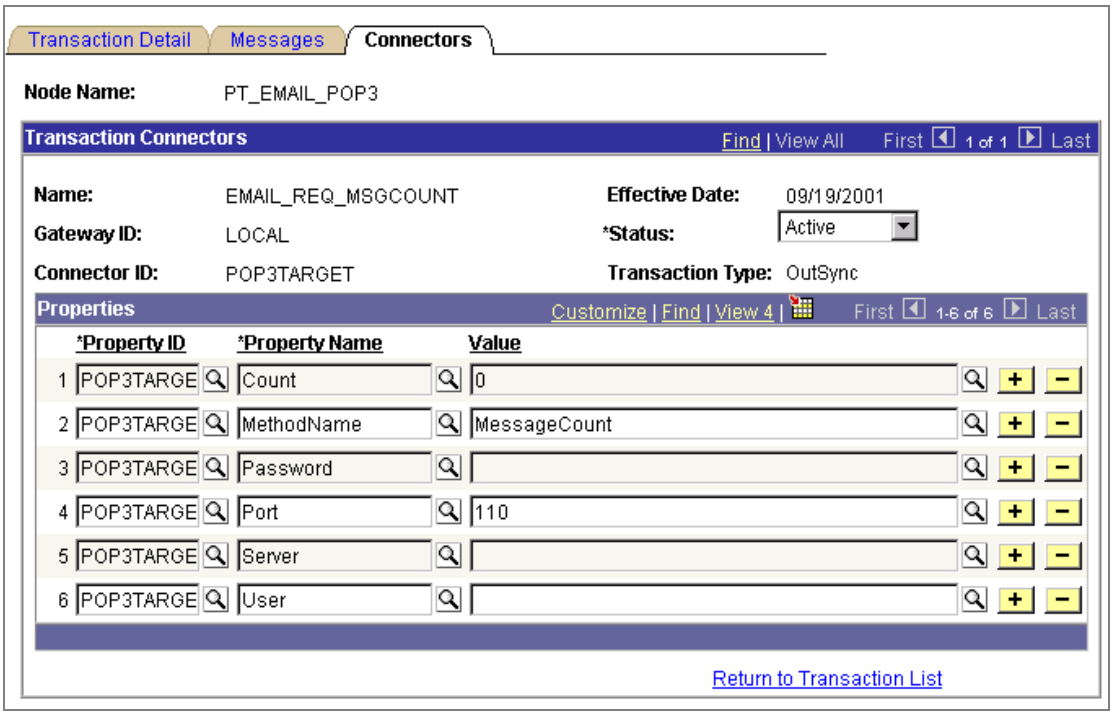
To set up transactions for PT_EMAIL_POP3

1. Select PeopleTools, Integration Broker, Node Definitions. The Nodes page appears.
2. Search for and open the POP3 default node, PT_EMAIL_POP3.
3. Click the **Transactions** tab. The pre-defined transactions for the node appear.



Transactions are pre-defined and loaded for the default PT_EMAIL_POP3 node.

4. Click the **Edit** link next to the transaction to set up. Transaction detail tabs appear.
5. Click the **Connectors** tab.



Required transaction properties for the EMAIL_REQ_MSGCOUNT transaction.

Note. You may have to click the **View All** link to view all properties for a transaction.

6. Specify **Values** for the following properties:
   - Password
   - Port
   - Server
   - User
7. Click **Save**.
8. To set up other transactions for the node, click the **Return to Transaction List** link and repeat steps 4 through 7.

Using the PT_EMAIL_POP3 Node

PeopleSoft provides a default node for use with the POP3 Target Connector called PT_EMAIL_POP3.

PeopleSoft provides the following message definitions with the PT_EMAIL_POP3 node. The **request message name** identifies a transaction for the node, The **response message name** indicates the response returned by the POP3 Target Connector. For example, when you send the message EMAIL_REQ_MSGCOUNT, it internally executes the service MessageCount, and returns the response in the EMAIL_RES_MSGCOUNT message.

The values each message returns depends on the service. The POP3 services are described earlier in this section.

Requesting Message Name	*Response Message Name*
EMAIL_REQ_MSGCOUNT	EMAIL_RES_MSGCOUNT
EMAIL_REQ_READALL	EMAIL_RES_READALL
EMAIL_REQ_READDELALL	EMAIL_RES_READDELALL
EMAIL_REQ_READMSG	EMAIL_RES_READMSG
EMAIL_REQ_DELMSG	EMAIL_RES_DELMSG
EMAIL_REQ_READHDR	EMAIL_RES_READHDR
EMAIL_REQ_READHDRATT	EMAIL_RES_READHDRATT
EMAIL_REQ_READMSGATT	EMAIL_RES_READMSGATT

All pre-defined request messages contain the **EMAIL_REQ_WRK record**. When a request message goes through the POP3 Target Connector, it retrieves data from the following fields in the EMAIL_REQ_WRK record:

Field	*Data Type*
User	CHAR (254)
Password	CHAR (254)
Server	CHAR (254)
Port	INT
MethodName	CHAR (16)
NUMROWS	INT
UIDL	CHAR (70)

The POP3 Target Connector then takes the field values from the EMAIL_REQ_WORK and generates a response. The response is contained in the **EMAIL_RES_WRK record**. The EMAIL_RES_WRK record contains the field values shown:

Field	*Data Type*	*Description*
EMAIL_FROM	CHAR (254)	From.
EMAIL_SENDER	CHAR (254)	Reply to.
UIDL	CHAR (70)	Unique identifier for the email message.
WL_SUBJECT	CHAR (255)	Subject.
EMAIL_TEXTLONG	LONG (512)	Body.
DATE_FROM	DATE	(RECV_DATE)

Field	Data Type	Description
NUMROWS (Count)	INT	Contains the count of messages. Used by the EMAIL_REQ_MSGCOUNT and EMAIL_REQ_DELMSG services.
NOTIFY_TO	LONG Use ' ; ' to separate list	To address collection
NOTIFY_CC	LONG Use ' ; ' to separate list	Cc address collection
ATTACHMENTS	LONG Use ' ; ' to separate list	Attachment list

There is an additional record returned when there is an attachment. The name of this record is **EMAIL_ATT_WRK**, which is a child of EMAIL_RES_WRK. EMAIL_ATT_WRK contains the following fields.

Field	Data Type
CONTENT_TYPE	CHAR (254)
FILENAME	CHAR (80)
FILE_DATA	ATTACHMENT
EMAIL_TEXTLONG	LONG

Retrieving Specific Messages from the POP3 Server

You can retrieve specific messages if you are using the ReadMessage, DeleteMessage and ReadMessageWithAttach services, To retrieve specific messages for these services, set the UIDL in the SyncRequest ( ) to the value of the message you want to read. The messages you specify are located in the EMAIL_RES_MESSAGE table.

If you want all messages returned, enter the value of *New* for the UIDL.

Code Examples – Passing Defined Node-Level Connector Properties

The following code examples demonstrate how to populate the UIDL field before the SyncRequest ( ) call to ensure that the PeopleSoft Integration Broker provides a message response in structured message format, when no field values are set in the request.

The examples also show the code you need to add to the end of the request to pass property values via method field records.

MessageCount

The following example shows how to execute SyncRequest for MessageCount:

```
Local Message &MSG;
Local Message &response;
```

```
&MSG = CreateMessage(Message.EMAIL_REQ_MSGCOUNT);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField
(Field.UIDL).Value = "NEW";

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     &str = &response.GetRowset().GetRow(1).GetRecord
     (Record.EMAIL_RES_WRK).GetField(Field.NUMROWS).Value;
End-If;
```

ReadAll

The following example shows how to execute SyncRequest for ReadAll:

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_READALL);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField
(Field.UIDL).Value = "NEW";

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get field values here */
End-If;
```

ReadMessage

The following example shows how to execute SyncRequest for ReadMessage:

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_READMSG);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.UIDL
).Value = &uidl ; /* UIDL of the message to read */

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get field values here */
End-If;
```

DeleteMessage

The following example shows how to execute SyncRequest for DeleteMessage:

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_DELMSG);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.UIDL
).Value = &uidl ; /* UIDL of the message to delete */

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get status of deleted message in NUMROWS field. */
     /* this is 1 if the message was found and deleted. */
End-If;
```

ReadHeaders

The following example shows how to execute SyncRequest for ReadHeaders:

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_READHDR);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.UIDL
).Value = "NEW";

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get field values here */
End-If;
```

ReadHeadersWithAttach

The following example shows how to execute SyncRequest for ReadHeadersWithAttach:

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_READHDRATT);
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.UIDL
).Value = "NEW";

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get field values here */
End-If;
```

ReadMessageWithAttach

```
Local Message &MSG;
Local Message &response;

&MSG = CreateMessage(Message.EMAIL_REQ_READMSGATT);

/* read value from the page */
&str = GetRecord(Record.EMAIL_RES_WRK).GetField(Field.UIDL).Value;
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField
     (field.UIDL).Value = &str;

&response = &MSG.SyncRequest(Node.PT_EMAIL_POP3);

If (&response.ResponseStatus = 0) Then
     /* get field values here */
     /* level 0 Rowset */
     &rs = &response.GetRowset();

     /* There is always an empty record, so start from 2 */
     For &i = 2 To &rs.ActiveRowCount

            /* Get each row */
            &row = &rs.GetRow(&i);
            /* level 0 record */
            &rec = &row.GetRecord(Record.EMAIL_RES_WRK);

            &rs1 = &row.GetRowset(Scroll.EMAIL_ATT_WRK);
            &count = &rs1.ActiveRowCount;
            For &j = 1 To &count
                   &row1 = &rs1.GetRow(&j);
                   &rec1 = &row1.GetRecord(Record.EMAIL_ATT_WRK);
                   &str = &rec1.GetField(Field.FILENAME).Value;
                   &BI_FILE.WriteLine("Message Part: " | &j);

                   /* Do different action based on Content Type */
                   &BI_FILE.WriteLine("Content Type:" |
                          &rec1.GetField(Field.CONTENT_TYPE).Value);

                   &str = &rec1.GetField(Field.FILENAME).Value;
                   &BI_FILE.WriteLine("File Name:" | &str);

                   /* Found a file name? - do something with the file */
                   /* data */
                   If All(&str) Then
```

```
                        /* if the file name was not available, */
                        /* Content-Id in the message is used, this
                        /* has "<" and ">" in it.

                        &str = Substitute(&str, "<", "");
                        &str = Substitute(&str, ">", "");
                        &FILE = GetFile("C:\TEMP\download\FILE" | &str,
                               "w", "a", %FilePath_Absolute);

                        &FILE.WriteRaw(&rec1.FILE_DATA.Value);
                        &FILE.Close();
                  Else
                        &BI_FILE.WriteLine("File Data: " |
                               &rec1.GetField(Field.EMAIL_TEXTLONG)
                               .Value);
                  End-If;
            End-For;

      &BI_FILE.WriteLine("Message " | &i);

      &str = &rec.GetField(Field.EMAIL_FROM).Value;
      &BI_FILE.WriteLine("From : " | &str);

      &str = &rec.GetField(Field.DATE_FROM).Value;
      &BI_FILE.WriteLine("Recv Date: " | &str);

      &str = &rec.GetField(Field.UIDL).Value;
      &BI_FILE.WriteLine("UIDL: " | &str);

      &str = &rec.GetField(Field.WL_SUBJECT).Value;
      &BI_FILE.WriteLine("Subject : " | &str);
      &str = &rec.GetField(Field.EMAIL_TEXTLONG).Value;
      &BI_FILE.WriteLine("Body: " | &str);
      &str = &rec.GetField(Field.ATTACHMENTS).Value;
      &BI_FILE.WriteLine("Attachments list : " | &str);
      End-For;
Else
      &BI_FILE.WriteLine("Bad Response");
End-If;
```

Code Examples – Overriding Defined Node-Level Connector Properties

You can override all the values defined for the POP3 Target Connector in the Node Definition at runtime by passing property values via method field records. The code for each service is the same as described in the previous section. However, you must also pass the following information: POP3 server name, user, password and method name.

To pass property values via method field records, you can take any of the code examples in the previous section, and add the following code to the end of each example:

```
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.SERV
ER).Value = <server name of POP3 server to connect>;
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.USER
).Value = <user name for the mail a/c to get emails>;
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.PASS
WORD).Value = <password for the user used>;
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.Meth
odName).Value = <Service name to be executed>;
/* must specify the port number */
&MSG.GetRowset().GetRow(1).GetRecord(Record.EMAIL_REQ_WRK).GetField(Field.Port
).Value= 110 ;
```

Using the Simple File Target Connector

The **Simple File Target Connector** allows you to write integration messages to a file in XML format, thus enabling you to verify that:

- You have composed messages correctly.
- The messages contain the content that you want to send.

The name of the output file takes the format:

```
<sourcenodename>.<messagename>.<publicationid>.xml
```

By default, the PeopleSoft Integration Broker writes the file to the C:\temp directory.

Node-Level Connector Properties

Set the following Simple File Target Connector properties in the Node Definition component when you set up the connector for an application.

Property ID	*Property Name*	*Description*
HEADER	SendUncompressed	Enables you to send messages decompressed. Values are: • *Y* (Default) • *N*
PROPERTY	FilePath	Indicates the location where the PeopleSoft Integration Broker writes the output file. The default location is *c:\temp*.
PROPERTY	FileName	(Optional) Indicates the name to use to overwrite the default file name.

Working with Listening Connectors

This section discusses:

- The flow of messages through listening connectors on the Integration Gateway.
- HTTP Listening Connector.
- PeopleSoft Listening Connector.
- PeopleSoft 8.1 Listening Connector.
- FTP Listening Connector.
- SMTP Listening Connector.
- JMS Listening Connector.
- POP3 Listening Connector.
- Simple File Listening Connector.

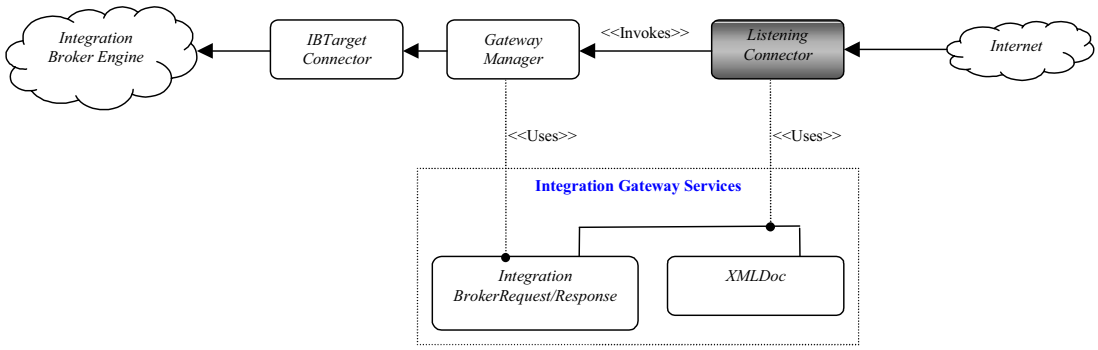
Understanding Listening Connectors

This section discusses:

- The flow of messages through listening connectors on the Integration Gateway.
- PeopleSoft-delivered listening connectors.

Listening connectors receive message requests from integration participants, send them to the Gateway Manager, and deliver responses back to the integration participants.

The following graphic shows the flow of an inbound message from an external system into the Integration Engine via a listening connector.



Message flow through a listening connector on the Integration Gateway.

PeopleSoft-Delivered Listening Connectors

PeopleSoft delivers four listening connectors with the PeopleSoft Integration Broker that enable integration participants to communicate with your PeopleSoft system using a number of communication formats. The listening connectors that PeopleSoft delivers with the PeopleSoft Integration Broker are:

- HTTP Listening Connector
- PeopleSoft Listening Connector
- PeopleSoft 8.1 Listening Connector
- JMS Listening Connector

All of the delivered listening connectors that service HTTP requests run as servlets and are configured to run in WebLogic or WebSphere web server environments. These connectors are: the HTTP Listening Connector, the PeopleSoft Listening Connector and the PeopleSoft 8.1 Listening Connector.

Using the HTTP Listening Connector

The HTTP Listening Connector monitors specific ports for incoming HTTP(S) messages intended for the Integration Engine. It is implemented as a Java HTTPServlet object.

This section describes the required and optional properties that messages bound for the HTTP Listening Connector pass. This section also provides examples of how to pass the credentials to the connector.

HTTP Listening Connector Properties

You must specify several required properties in messages that you pass through the HTTP Listening Connector. There a several optional properties you can specify as well.

Specify HTTP Listening Connector properties in the HTTP header and in the URL query string specification.

These properties, also known as credentials, are meta data that are specific to each message that the HTTP Listening Connector processes. These properties supply authentication information and descriptive details about how the message is to be processed. For each message that is sent to the connector, the PeopleSoft Integration Broker uses the property values that you supply to create an IBRequest that it uses to process and service the request internally.

Specify the following required and optional properties in the HTTP message header.

Requestor Node	Identifies the name of the node that is making the request. This field is optional if you are using a distinguished name with a digital certificate.
Message Name	Identifies the name of the message.
Requestor Password	(Optional) If password authentication is used, you must enter the password as entered in the node definition. This entry enables password authentication to allow processing of the message.
Destination Node	(Optional) Identifies the name of the node that will receive the message.

Message Version	(Optional) Identifies the name of the node that will receive the message.
Originating User	(Optional) Identifies the user ID login from which the message was initially generated.
Originating Node	(Optional) Identifies the name of the node that started the process.
Originating Process	(Optional) Identifies the name of the process from which the publish originated. For example, a message published from the Inventory definitions page has a process name of *INVENTORY DEFIN*.
Originating Timestamp	(Optional) Identifies the time at which the original request was created.
Message Type	(Optional) Identifies the type of message being sent. Values are: • *Sync:* Specifies that the message that you are sending is synchronous. • *Async:* Specifies that the message that you are sending is asynchronous. • *Ping:* Specifies that the message that you are sending is intended to attain the active or inactive status of the destination node.
Final Destination	(Optional) Identifies the name of the node that will ultimately receive the message. This is common in cases in which a PeopleSoft Integration Broker Hub is used.
Non-Repudiation	(Optional) Specifies whether the message content in the request should be processed using nonrepudiation logic. Values are: • *Y:* Nonrepudiation logic is to be used. • *N:* Nonrepudiaton logic should not be used.

Passing Properties to the HTTP Listening Connector

You can pass properties to the HTTP Listening Connector in:

- The body of a message in the supported PeopleSoft XML Envelope format through an HTTP Post.
- The URL query string through an HTTP Get or Post.
- The HTTP header through an HTTP Get or Post.

Using an HTTP Post is the *only* way in which you can send the body of a message to the PeopleSoft Integration Broker using the HTTP Listening Connector.

However, you can use an HTTP Get in cases when you do not need to post the body of a message. In this case, you pass the HTTP Connector properties in the URL query string or in the HTTP header. For instance, you might have requests for information, (queries), such as a request for a customer list. In this case, you need only specify the message name (for example, CUSTOMER_LIST_REQUEST), the requesting node and possibly some other property in the URL query string of the HTTP header.

The following code example shows the format for sending HTTP Listening Connector properties through the PeopleSoft XML Envelope format:

```
<?xml version="1.0"?>
     <IBRequest>
            <MessageName/>
            <MessageType/>
            </From>
            <RequestingNode/>
            <Password/>
            <OrigUser/>
            <OrigNode/>
            <OrigProcess/>
            <OrigTimeStamp/>
     </From>
     <To>
            <FinalDestination/>
            <DestinationNode/>
     </To>
     <ContentSections>
            <ContentSection>
                  <NonRepudiation/>
                  <Data/>
            </ContentSection>
     </ContentSections>
</IBRequest>
```

The following example shows the format for passing HTTP Listening Connector properties in the URL query string. The header must include the required properties, requesting node and message name.

The URL query string format is:

```
http://<machine name>:<port>/PSIGW/HTTPListeningConnector?MessageName=<name of
message>&From=<requesting node>&Password=<requestor
password>&To=<targetentity>&OrigUser=<originating
user>&OrigNode=<originatingnode>&OrigProcess=<originating
process>&OrigTimeStamp=<original time request was
created>&MessageType=<Sync/Async/Ping>&FinalDestionation=<ultimate target of
message>NonRepudiation=<boolean>
```

The following example shows the format for passing HTTP Listening Connector properties through the HTTP header. The header must include the required properties, requesting node and message name.

The HTTP header format is:

```
From: =<requesting node>
Password: <requestor password>
MessageName: <name of message>
MessageType: <Sync/Async/Ping>
To: <destination node>
FinalDestination: <ultimate destination node>
OrigUser: <originating user>
OrigNode: <originating node>
OrigProcess: <originatingprocess>
OrigTimeStamp: <original time request was created>
NonRepudiation: <boolean>
```

Responses to XML Envelope, HTTP Header and URL Query String Methods

For messages of type *asynchronous*, the Integration Gateway immediately returns an XML document to the sender in one of two forms:

- If successful, an asynchronous acknowledgement response.
- If unsuccessful, an error response.

For messages of type *synchronous*, the Integration Gateway returns a response to the sender in one of two forms:

- If successful, a response in a format determined by the transaction properties for the node (for example the message, transform and so forth).
- If unsuccessful, an error response contained in an XML document.

The following example shows a successful asynchronous acknowledgement response:

```
<?xml version="1.0"?>
<IBResponse type="success">
            <DefaultTitle>Integration Broker Response</DefaultTitle>
            <StatusCode>0</StatusCode>
<TransactionID>UNDERDOG.QE_SALES_ORDER_ASYNC_CHNL.20</TransactionID>
</IBResponse>
```

The HTTP Listening Connector uses a standard error XML format for all requests, except for SOAP requests, if error handling is invoked in the Integration Gateway. The following is an example of this standard error response:

```
<?xml version=”1.0”?>
     <IBResponse type=”error”>
            <DefaultTitle>Integration Broker Response</DefaultTitle>
```

```
            <StatusCode/>
            <MessageID/>
            <DefaultMessage/>
            <MessageParameters>
                  <Parameter/>
            </MessageParameters>
</IBResponse>
```

Cookie Support

The HTTP Listening Connector supports cookies. Cookies that are passed as part of a message request to the HTTP Listening Connector are processed, read and manipulated at the PeopleCode level in the Integration Engine. You enter Cookies in the message header.

The following is an example of a Cookie.

```
Cookie: foo=bar; path=/; expires Mon, 09-Dec-2002 13:46:00 GMT
```

In this example, the header entry would result in a cookie named *foo*. The value of *foo* is *bar*. This cookie has a path of /, meaning that it is valid for the entire site, and it has an expiration date of Dec 9, 2002 at 1:46pm Greenwich Mean Time (or Universal Time). Provided the browser can understand this header, the cookie will be set.

SOAP Message Support

To support SOAP (Simple Object Access Protocol) messages, HTTP Listening Connector properties are passed in a SOAP-specific header.

An inbound SOAP request that contains the message name and HTTP Listening Connector properties must be in a SOAPAction HTTP header. The value that you pass in this header must be a string that is formatted as follows:

```
#MessageName#RequestingNode#Password.
```

The following example shows a sample SOAPAction HTTP header:

```
POST /get_BindingDetail HTTP/1.1
Host: www.someOperator.com
Content-Type: text/xml; charset="utf-8"
Content-Length: nnnn
SOAPAction:#PURCHASE_ORDER#PSFT#PSFTPassword
```

If an error occurs in the Integration Gateway during processing, a SOAP-specific XML error is generated instead of a standard XML error. The following example shows an example of an error in SOAP-specific XML format:

```
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
     <SOAP-ENV:Body>
            <SOAP-ENV:Fault>
                  <faultcode>SOAP-ENV:Server</faultcode>
```

```
                    <faultstring>Server Error</faultstring>
                    <detail>
                           <IBResponse type="error">
                                  <DefaultTitle>
                                  Integration Broker Response
                                  </DefaultTitle>
                                  <StatusCode>10</StatusCode>
                                  <MessageID>10731</MessageID>
                                  <DefaultMessage></DefaultMessage>
                           </IBResponse>
                    </detail>
             </SOAP-ENV:Fault>
      </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Using the PeopleSoft Listening Connector

The PeopleSoft Listening Connector is used by the Integration Engine to communicate with the Integration Gateway. It is also used to receive requests from Integration participants in the PeopleSoft internal messaging format.

Like the HTTP Listening Connector, the PeopleSoft Listening Connector is implemented as a Java HTTPServlet object. However, the PeopleSoft Listening Connector expects its request to be in PeopleSoft MIME format.

The Integration Engine sends messages formatted in MIME over HTTP. The PeopleSoft Listening Connector receives these messages as Posts (Get requests cannot be made in this way) and immediately converts the MIME input into a Java string object.

The PeopleSoft Listening Connector logs these requests and then invokes the Gateway Manager to unmarshall the string into an IBRequest object. The Gateway Manager invokes the target connector specified in the ConnectorClassName field in the IBRequest, which is derived from the node definition on the source Integration Engine.

The Gateway Manager returns the responses to the PeopleSoft Listening Connector, where they are logged and sent back to the original requesting systems, typically Integration Engines.

Using the PeopleSoft 8.1 Listening Connector

The PeopleSoft 8.1 Listening Connector enables you to receive messages from PeopleSoft 8.1 systems.

In PeopleSoft 8.1 systems, Application Messaging generates highly structured XML messages that are designed to be sent to PeopleSoft 8.1 Application Messaging Gateways.

The PeopleSoft 8.1 Listening Connector mimics the role of the Application Messaging Gateway by receiving and processing PeopleSoft 8.1 messages. However, this connector transforms the PeopleSoft 8.1 messages into a PeopleSoft 8.4-formatted XML message that

can be processed by the Integration Gateway and ultimately by the Integration Engine. This conversion is necessary because the two message formats are distinctly different.

Using the JMS Listening Connector

The JMS Listening Connector has two components.

- A subscriber.
- A queue listener.

The JMS **Subscriber** subscribes to different topics and the JMS **Queue Listener** polls message queues for new messages.

The JMS Listening Connector features a Subscription Manager that retrieves the topics to which it subscribes and the Queues to which it must listen.

Note. The JMS Listening Connector always expects JMS messages to be in text format.

If the JMS Listening Connector receives a message that it cannot process or if the PeopleSoft Integration Broker generates an exception, the connector publishes the message to the Error Topic. You must set up the Error Topic and its properties in the IntegrationGateway.properties file.

The JMS Listening Connector retrieves different Topics and Queues from the IntegrationGateway.properties file. For each topic, it starts a Topic Subscriber and for each Queue it starts a Queue Listener. When a message arrives either for a Queue or Topic, the JMS Listening Connector sends the message to the Peoplesoft Integration Broker. If an error occurs while sending the message to the Peoplesoft Integration Broker, it publishes an error or message to the Error Topic.

When a message arrives at the Integration Gateway, the connector looks for various string properties in the JMS Message header and routes the message to the proper destination. If any of the properties are missing, the connector publishes an error to the Error Topic.

Note. If the application server returns a Status 20, the message is published to the Error Topic and the response is logged in the Integration Gateway message log.

The JMS Listening Connector provides synchronous and asynchronous messaging. It determines whether the message is a Request-Reply (asynchronous message) by using the JMS Message header property ReplyTo. When this property is *True* the response is sent back on the Temporary Queue or Topic. When the property is *False* it simply consumes the message and does not send a reply to the requestor.

When the message received specifies a mode of *Request-Reply,* a reference to the Temporary Queue or topic must be set in the JMS Message header for the JMS Listening Connector to know where to send the message reply. The JMS Listening Connector also sets the JMS correlation ID when it is sending back the replies so that the requestor can properly associate the reply with its corresponding request.

Starting the JMS Listening Connector

You can start the JMS Listening Connector using the JMSListeningConnectorAdministrator servlet, or you can start it manually.

To start the JMS Listening Connector using a servlet

1. Deploy the servlet under Web Application PSIGW.
2. To start the servlet on start up of the Web server, set the variable *Load On Startup*.

   When you set the Load On Startup option, the JMS Listening Connector automatically starts when you start your Web server. Refer to your Web server documentation for more details about starting your servlet automatically when your web server starts.

To start the JMS Listening Connector manually

1. Open a browser and enter the following URL:

   http://localhost/PSIGW?JMSListeningConnectorAdministrator?Activity=Start

2. Press ENTER.

Shutting Down the JMS Listening Connector

You can shut down the JMS Listening Connector by stopping the JMSListeningConnectorAdministrator servlet.

To shut down the JMS Listening Connector

1. Open a browser and enter the following URL:

   http://localhost/PSIGW?JMSListeningConnectorAdministrator?Activity=Stop

2. Press ENTER.

Note. You must register JMSListeningConnectorAdministrator servlet object under the web application PSIGW in the web server. Your WebLogic or WebShpere administration manual should provide you with instructions about how to register the servlet.

Message Header Properties

You must set the following properties in the message header when you publish messages from a JMS system to the PeopleSoft Integration Broker.

MessageName	Identifies the name of message.
RequestingNode	Identifies the requesting node name.

FinalDestinationNode	Identifies the final destination nodes. If there are no values, set to *Null.*
DestinationNode	Identifies the destination node names, separated with commas. If there are no values, set to ″ ″ (empty string).
Password	Identifies the password to the system.
ReplyTo	Identifies whether a reply is required. Values are: • *True:* A reply is required. Sets up synchronous publishing. • *False:* A reply is not required. Sets up asynchronous publishing.

When any of the message properties are not set an error is logged and an error is published to the Error Topic. The message that the connector publishes to the Error Topic has a property call Error and is set to True. The error message that is published contains the following information: default message, message ID, message set, message parameters and body of the message sent.

Gateway-Level Connector Properties

To use the JMS Listening Connector, you must set several properties in the IntegrationGateway.properties file. See Setting Integration Gateway Properties

Required Java Archive (JAR) Files

When you use the JMS Listening Connector, specific JAR files are required based on the web server that you are using.

WebLogic	Weblogic.jar
iPlanet	jms.jar, ,jmq.jar, fscontext.jar, jndi.jar
MQSeries	jms.jar, ,jndi.jar, fscontext.jar. com.ibm.mqjms.jar

Setting Up Gateway Certificates

If you send information over the Internet, the recipients of that information might need to verify that the information came from you and that it was not modified during the send. Gateway certificates help to secure the information that you send over the Internet.

For the PeopleSoft Integration Broker, you can set up gateway certificates when you are implementing integrations with PeopleSoft systems using remote gateways, or when you are implementing integrations with third-party systems.

Understanding Gateway Certificates

Certificates are comprised of three components:

- Digital signatures.
- Private keys.
- Public keys.

You can digitally "sign" messages using a **private key**. (To generate a private key, use the security API method of your choice.) When you generate a private key, a **public key**, which enables the message recipient to verify that the message came from you, is also generated.

When you send message with gateway certificates defined, the Integration Gateway sends the message and the certificate. The recipient can check whether the certificate is valid is by verifying the signature using the public key.

A collection of certificates for entities that you trust is typically imported into a keystore that is located on the web server. A **keystore** is a flat file that contains trusted certificates and combinations of private keys, and their corresponding certificates. Each certificate uses a **certificate alias** name to identify itself inside the keystore.

Setting Up Gateway Certificates

Set up Integration Gateway certificates in the IntegrationGateway.properties file. When you set up certificates, you specify the following properties in the file:

- Certificate alias and password.
- Path to the keystore and the keystore password.

Before you enter this information, you must encrypt the certificate alias password.

This section discusses how to:

- Encrypt the certificate alias password.
- Set up Integration Gateway certificates.

Encrypting the Certificate Alias Password

PeopleSoft provides a utility to encrypt the password. After you encrypt the password, you need only copy and paste the password into the IntegrationGateway.properties file.

You only provide an encrypted password for the certificate alias; you do not encrypt the keystore password.

To encrypt the certificate alias password

1. Open a DOS Command prompt.
2. Go to $PS_HOME/class (the location of the gateway installation).
3. Enter the following command:

```
Java psft.pt8.pshttp.PSCipher <yourpassword>
```

4. Press **ENTER**.

An encrypted password appears. This is the password that you cut and paste as the certificate password in the IntegrationGateway.properties file.

Setting Up Gateway Certificates

To set up Integration Gateway certificates

1. Open the IntegrationGateway.properties file in a text editor.
2. Find the Certificates Section of the file.
3. Set the ig.certificateAlias property to the certificate alias name.
4. Set the ig.certificatePasswd property to the password that you encrypted in the previous section.
5. Set the SecureFileKeystorePath to the location of the pskey file.

   The location varies, depending on the web server.

   WebLogic:

   - c:/bea/wlserver6.1/config/peoplesoft/keystore/pskey

   WebSphere:

   - c: /WebSphere/AppServer/installedApps/peoplesoft/keystore/pskey
6. Set the secureFileKeystorePasswd property to the Keystore password.
7. Save the file.
8. Refresh the Integration Gateway.

Note: The certificate is not in effect until you set the Authentication option in the Node Definition component.

See Also

Setting Integration Gateway Properties

"Understanding Integration Broker," Integration Gateway Architecture

"Administering Basic Integrations," Refreshing the Gateway Properties

Running the Integration Gateway Behind a Proxy Server

When a proxy server is setup for a network on which the Integration Gateway resides, all HTTP transactions are routed through that proxy server automatically. The HTTP transport layer uses the proxy server settings from the IntegrationGateway.properties file. The message or transaction is routed to the proxy server and then on to the Internet.

Only the HTTP Target Connector can use a proxy server, if set. The other connectors are not affected.

This section provides an overview of proxy authentication and discusses how to set gateway-level properties.

Understanding Proxy Authentication

The Authentication (proxy) header is an HTTP header used to authenticate the user with a user ID and password. You set the user ID and password in the Proxy-Authentication header on an HTTP transaction. For the Integration Gateway, you must set the value of the Proxy Authentication header to *userid:password*. The HTTP Target Connector does a Base64 encoding of the header and sets the header value adding the proper encoding instructions. This header is then used as the challenge by the proxy server, and verifies the user ID and password with its own sign-on data.

Setting Gateway-Level Connector Properties

To run the Integration Gateway behind a proxy server, you must uncomment and add values for the properties in the Proxy Server Settings section of the IntegrationGateway.properties file.

Proxy Host Indicates the domain name of the proxy server.

Proxy Port Indicates the port number of the proxy server.

See Also

Setting Integration Gateway Properties

CHAPTER 13

Using the Integration Broker Connector SDK

PeopleSoft bundles connectors with its Integration Broker product for use with PeopleSoft, HTTP, JMS, PeopleSoft 8.1, FTP, POP3 and SMTP communication formats. PeopleSoft also provides an Integration Broker Connector Software Development Kit (SDK) that you can use to build and implement connectors for other communication formats and application requirements.

This chapter discusses how to:

- Develop and implement custom connectors.
- Develop connectors based on DOM.
- Develop connectors for the C/C++ environment.
- Reuse connector code.
- Test message processing using Send Master.
- Post third-party messages to the Integration Gateway using the Simple Post Tool.

Understanding the Integration Broker Connector SDK

The Integration Gateway exposes a variety of methods to extend itself to support functionality related to new connectors. Generally, you create new connectors when you need a connection to an external system that you cannot achieve with the connectors delivered with Integration Broker.

For example, you could develop a connector for integrations with a PeopleSoft Human Resources system to an SAP Financials system, a PeopleSoft Supply Chain system to a supplier using a RosettaNet system, a PeopleSoft eProcurement system to a Commerce One Marketsite using an XCBL / SAX system.

This section discusses:

- The contents of the Integration Broker Connector SDK.
- The location of the Integration Broker Connector SDK.
- Integration Broker Connector SDK API documentation.

Integration Broker Connector SDK Contents

The following items are included in the Integration Broker Connector SDK:

- Instructions for setting up your development environment.
- Java classes required for creating connectors (including IBResponse and IBRequest objects).
- API documentation.
- Sample code for listening and target connector classes.
- Send Master Post utility to test connectors.
- Simple Post utility that allows third-party systems to post messages to the Integration Gateway.

Integration Broker Connector SDK Location

The location of the SDK and its contents depend on the Web server you are using.

WebLogic Web Server

SDK	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Sdk
Instructions for setting up your connector development environment.	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Sdk\ docs\ReadMe.txt
Java Classes	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\ WEB-INF\classes
API Documentation	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\ Sdk\docs\index.html
Sample code for listening and target connector classes	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\ Sdk\src
Send Master	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW
Simple Post Tool	c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\WEB-INF\classes\com\peoplesoft\pt\simplepost

WebSphere Web Server

SDK	c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Sdk
Instructions for setting up the connector development environment.	c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Sdk\docs\ReadMe.txt

Java Classes	c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\WEB-INF\classes
API Documentation	c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Sdk\docs\index.html
Sample code for listening and target connector classes	c:\websphere\AppServer\installedApps\peoplesoft\PSIGW\Sdk\src
Send Master	c:\websphere\InstalledApps\peoplesoft\PSIGW
Simple Post Tool	c:\websphere\InstalledApps\peoplesoft\PSIGW\WEB-INF\classes\com\peoplesoft\pt\simplepost

Integration Broker Connector API Documentation

The Integration Broker Connector SDK includes API documentation in HTML format.

The documentation lists and describes Java packages you can use to develop custom connectors, and includes information about package classes, inner classes, interfaces, constructors, methods and fields.

You can access the Connector API documentation in the Integration Broker SDK directory

Understanding Connector Development and Implementation

There are differing ways to produce new connectors based on whether you want to create a listening connector or a target connector.

Listening connectors use a standard connector interface to the Integration Gateway, as well as Gateway Services to link between the new listening connector and the Integration Gateway. While a Java interface object is not used for listening connectors, they still must adhere to a standard scheme of logic in order to drive requests to and process responses from the Integration Gateway.

Target Connectors on the other hand do have to implement an actual Java interface in order to become valid target connectors in the Integration Gateway. This ensures a standard interface for the Gateway Manager so that it may manage each target connector in a streamlined fashion.

Developing and implementing custom connectors is a three-step process:

1. Develop a connector class.
2. Install the connector class.
3. Register the connector (target connectors only).

In addition to providing information on these topics, this section also includes information about functionality that you should incorporate into all connectors, as well as information you should implement in target connectors or listening connectors.

Included in the Integration Broker Connector SDK are Java classes, API documentation and sample code that you can use for your development efforts. In addition, target and listening connector templates are provided at the end of chapter for you to use as a starting point for connector development.

This section discusses:

- Connector development infrastructure.
- General connector class development considerations.
- Target connector class development considerations.
- Listening connector class development considerations.
- Installing connector classes.
- Registering connectors.
- Connector templates.

Connector Development Infrastructure

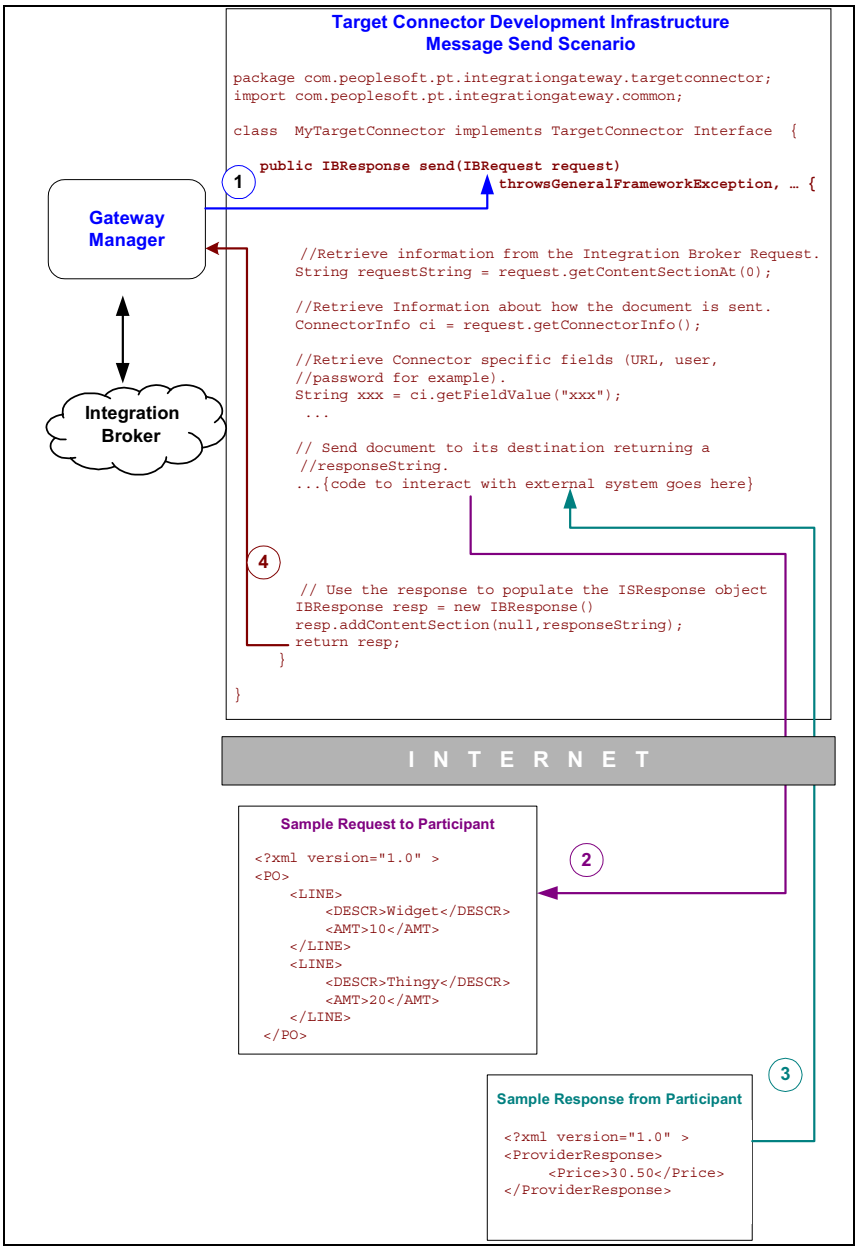
Target connectors generate message requests, send them to integration participants, wait for responses from participants and deliver the responses back to the Gateway Manager. Listening connectors receive message requests from integration participants, send them to the Gateway Manager, and deliver responses back to the integration participants.

This section discusses:

- Target connector development–Message send scenario.
- Target connector development–Ping scenario.
- Target connector development –Introspection scenario.
- Listening connector development infrastructure.

Target Connector Development Infrastructure—Message Send Scenario

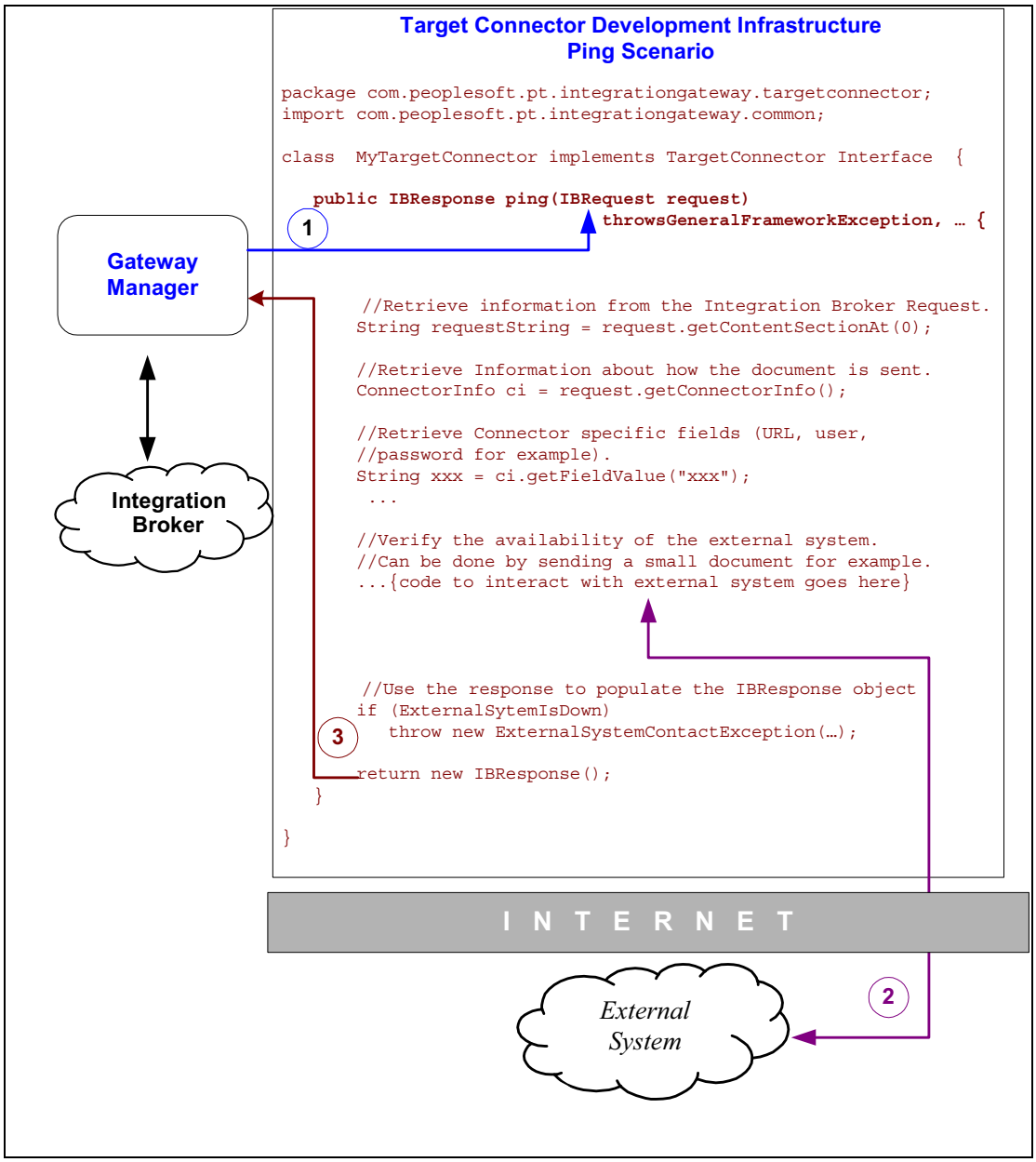
When you develop target connectors, you can develop them to generate and send message requests to integration participants and return responses The following graphic shows how the connector code accomplishes these tasks.



Target connector – message send scenario

Target Connector Development Infrastructure –Ping Scenario

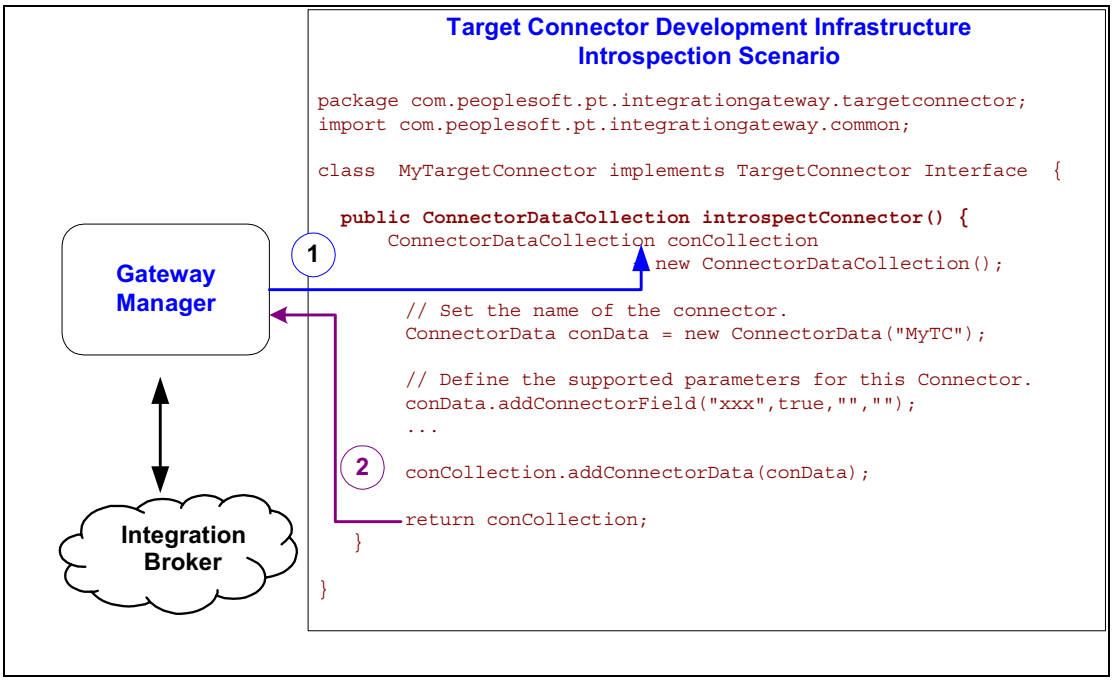
When you develop target connectors, you can also develop them to ping external systems. The following graphic shows how the connector code accomplishes this task.



Target connector– ping scenario

Target Connector Development Infrastructure –Introspection Scenario

When you develop target connectors, you can develop them to handle connector introspection. The following graphic shows how the connector code accomplishes this task.



Target connector – introspection scenario

General Connector Class Development Considerations

While implementations vary greatly, when you develop connector classes there is specific functionality that you should incorporate. This section describes that functionality and includes the following topics:

- Handling Specific Input and Output Formats Exchanged through Connectors.
- Handling the Interaction between Local and External Systems.

Handling Input and Output Formats Exchanged through Connectors

For a target connector to handle input and output formats exchanged with its intended recipient, it must transform the Integration Broker request (IBRequest) into a message formatted for the intended external system.

For instance, the HTTP Target Connector delivered with the Integration Broker gathers HTTP headers and cookies from the Integration Broker request and formulates the appropriate HTTP message, complete with the actual message content, so it can be delivered to its destination. When the response comes back, the connector creates an Integration Broker response (IBResponse) using the response string.

For a listening connector to handle input and output formats exchanged with its requestor, it must transform the incoming message into an Integration Broker request (IBRequest). For example, PeopleSoft built the HTTP Listening Connector it delivers with Integration Broker to recognize SOAP messages and to retrieve query string arguments, HTTP headers and cookies. It then formats all this information to create the appropriate request (IBRequest) so

that the Integration Broker can converse with it. When the response comes back, the HTTP Listening Connector reads the Integration Broker response (IBResponse) and sends its output message content back to the requesting system.

Handling the Interaction between Local and External Systems

A target connector interacts with an external system by sending it information and retrieving the response.

As an example, to accomplish this interaction the HTTP Target Connector delivered with the Integration Broker uses various HTTP-specific classes to handle sending messages over the HTTP protocol, as well as to handle situations such as the external system being down, security (through HTTPS), and so forth.

A listening connector interacts with an external system by receiving requests from the external system and returning responses that the external system understands. As an example, to accomplish this interaction the HTTP Listening Connector delivered with the Integration Broker uses a servlet to receive and reply to incoming HTTP messages.

Target Connector Class Development Considerations

This section describes target connector class development issues to consider when creating custom target connectors.

The section discusses:

- Target connector interface.
- Building introspection into target connectors.
- Building error handling and logging into target connectors.

Target Connector Interface

As with PeopleSoft-provided target connectors, the Integration Gateway dynamically invokes custom-built target connectors via the Gateway Manager.

Target connectors must adhere to a standard structure as defined in the Target Connector Interface.

```
public interface TargetConnector {
     IBResponse send(IBRequest request) throws
                                              GeneralFrameworkException,
                                              DuplicateMessageException,
                                              InvalidMessageException,
                                              ExternalSystemContactException,
                                              ExternalApplicationException,
                                              MessageMarshallingException,
                                              MessageUnmarshallingException;
```

```
    IBResponse ping(IBRequest request) throws
                                                GeneralFrameworkException,
                                                DuplicateMessageException,
                                                InvalidMessageException,
                                                ExternalSystemContactException,
                                                ExternalApplicationException,
                                                MessageMarshallingException,
                                                MessageUnmarshallingException;

ConnectorDataCollection introspectConnector();
```

You use the **Send** method to send a request to an external system and retrieve its response. The Gateway Mangager passes the request to this method and expects a response to be returned.

The **Ping** method enables the Integration Broker to verify the availability of a site. The Integration Broker Monitor can also invoke the Ping method explicitly.

ConnectorDataCollection invokes introspection.

The Integration Broker Connector API documentation contains more information about these methods.

You have the option of using Gateway Services such as XMLDoc for document access and mutation. You also have access to the Integration Broker Response (IBResponse) and Integration Broker Request (IBRequest) objects from the Integration Gateway.

Building Introspection into Target Connectors

The Integration Broker has the ability to introspect (or query) the capabilities of target connectors installed on the Integration Gateway (remote or local). All target connectors delivered with Integration Broker are loaded using introspection using the Load button located on Connectors page in the Gateways component.

You can build introspection into custom-built connectors. When you do so, you can load the connector and its properties with the click of a button.

In order for the introspection process to gather information about a custom target connector, you must supply information inside the target connector by implementing the **IntrospectConnector** method.

The following example is from the SMTP Target Connector that is responsible for sending messages via e-mail:

```
public ConnectorDataCollection introspectConnector() {
     //Creates the ConnectorDataCollection that will be returned
     //by this method. This object will contain all the
     //necessary information about this Connector's properties.

            ConnectorDataCollection conCollection = new
ConnectorDataCollection();
```

```
            //Create ConnectorData Object and stipulating the name of
            //the connector as seen from the Gateway Component.

            ConnectorData conData = new ConnectorData("SMTPTARGET");

            conData.addConnectorField("DestEmailAddress", true, "", "");
            conData.addConnectorField("SourceEmailAddress", true, "", "");
            conData.addConnectorField("CC", false, "", "");
            conData.addConnectorField("BCC", false, "", "");
            conData.addConnectorField("HEADER", Content-type", false,
                   "", "text/plain|text/html");
            conData.addConnectorField("HEADER","sendUncompressed",true,
                   "Y","Y|N");

            //Add the ConnectorData to your ConnectorDataCollection Object.
                         //Typically, you would only

            //add one ConnectorData into your ConnectorDataCollection.
            conCollection.addConnectorData(conData);

      return conCollection;
}
```

You use the following method to add connector fields.

addConnectorField *([PropertyID] PropertyName, Required, DefaultValue, PossibleValues)*

This method takes the following parameters:

Property ID	Identifies different property types, such as HEADER for HTTP or SMTP. PeopleSoft also uses the HEADER *PropertyID* to allow for the message to be sent in either compressed or clear format through the *sendUncompressed* property. If not supplied, the PropertyID will be equal to the *ConnectorName*.
Property Name	This is the name of the connector property.
Required	Determines if the information is required for the target connector to deliver its message. The valid values are: • *Y*: True. • *N*: False.
Default Value	The default value for the property. Typically, you will want to set the Required parameter to True when you specify a Default Value so that this information carries to the Node Configuration in the Integration Broker Engine.
Possible Values	A list of the possible values that the property can take, separated by the "|" character.

You can see how these properties function by walking through the following definition from the previous example:

```
conData.addConnectorField("HEADER","sendUncompressed",true,"Y, "Y|N");
```

In this case, the *PropertyName* is **sendUncompressed** and its *PropertyID* is **HEADER**. The **sendUncompressed** property is *Required* (as evidenced by the third parameter being set to true), which means that whenever you create a node in the node definition component and specify SMTPTARGET as your target-connector, this property will appear on your page automatically. Further, since the *DefaultValue* is set to **Y**, you will see this as the default value. The *PossibleValues* have been identified as **Y** or **N**. If you were to click on the list box (search box) for this property on the Connectors tab of the Node Definition component you would be able to select from those two values.

Building Error Handling and Logging into Target Connectors

This section provides a code example that demonstrates how to build error handling and logging into target connectors.

```
package com.peoplesoft.pt.integrationgateway.targetconnector;

import ...

public class SampleTargetConnector implements TargetConnector {
                  public IBResponse ping(IBRequest request)

                  public IBResponse send(IBRequest request)throws
                                          GeneralFrameworkException,
                                          InvalidMessageException,
                                          ExternalSystemContactException,
                                          ExternalApplicationException,
                                          MessageMarshallingException,
                                          MessageUnmarshallingException,
                                          DuplicateMessageException  {

            PSHttp httpObj = null;
            try {

            // Get handle on rootnode
            XmlNode root = dom.GetDocumentElement();

            // Cast the IBRequest back to an InternalIBRequest
            InternalIBRequest request = (InternalIBRequest)requestOrig;

            // Populate App Msg XML  Dom Object from IBRequest

            ...
```

```
        // Get the URL from either the IBRequest or from the
        //prop file (default)

        String URL = request.getConnectorInfo().getFieldValue("URL");

        // Log the request
        Logger.logMessage("SampleTargetConnector:
        Application Message Request", dom.GenerateXmlString(),
        Logger.STANDARD_INFORMATION);

        // Send the request DOM Document
        httpObj.setRequestProperty("content-type", "text/plain");
        httpObj.send(dom.GenerateXmlString().getBytes());

        // Get the response and convert to a String
        responseString = new String(httpObj.getContent());

        // Log the response
        Logger.logMessage("SampleTargetConnector:
        Application Message Response", responseString,
        Logger.STANDARD_INFORMATION);

        // Construct the IBResponse
        response = new IBResponse();

        ...

        // Return the successful IBResponse
        return response;

    } catch (XmlException xe) {
        httpObj.disconnect();
        throw new GeneralFrameworkException ("SampleTargetConnector:Failed
            while parsing XML");

    } catch (org.w3c.www.protocol.http.HttpException httpe) {
        throw new ("SampleTargetConnector:HTTP Protocol
            exception",httpe);


    } catch (java.io.IOException ioe) {
        throw new ExternalSystemContactException
            ("SampleTargetConnector:I/O Exception",ioe);

    } finally {
```

```
            httpObj.disconnect();
        }
        } // end send()
    }
```

Listening Connector Class Development Considerations

This section describes listening connector class development issues to consider when creating custom listening connectors, and discusses:

- Building servlet-based versus non-servlet-based listening connectors.
- Invoking listening connectors.
- Controlling message routing.
- Building error handling and logging into listening connectors.

Building Servlet-Based versus Non-Servlet-Based Listening Connectors

If you require a listening connector that services HTTP requests, you need to build a servlet-based listening connector. A servlet-based listening connector runs in the Servlet container on the Web server.

The Connector Templates topic in this section features a servlet-style listening connector template that you can use as a starting point for your connector development.

See the servlet documentation for your Web server for information about installing servlets on the Web server.

Invoking Listening Connectors

Listening connectors must invoke the Integration Broker through the Gateway Manager **Connect** method.

```
IBResponse connect(IBRequest) throws
     GeneralFrameworkException
     DuplicateMessageException
     InvalidMessageException
     MessageMarshallingException
     MessageUnmarshallingException
     ExternalSystemContactException
     ExternalApplicationException
```

Controlling Message Routing

By accessing and modifying key information on the IBRequest, you can control the behavior of transactions as they flow through the Integration Gateway.

This section describes several dispatching features you can use to control message routing by modifying the IBRequest from your listening connector, including routing messages to:

- Other (remote) Integration Gateways.
- Specific target connectors.
- Other PeopleSoft systems.

You can control the routing of your message to another Integration Gateway by specifying the URL of the Gateway in the IBRequest. Sometimes, you might need to forward messages to another Gateway so they can be processed by a remote Peoplesoft 8.4 system. In order to do this you can simply specify the URL of this Integration Gateway as follows:

```
…
IBRequest ibRequest = new IBRequest();
IbRequest.setMessageName("RemoteRoutingTest");
IbRequest.setRequestingNode("SourceSystem");
IbRequest.setPassword("myPassword");
…

// Specify the processing of the message to occur from //anotherIntegration
Gateway.
ibRequest.setRemoteFrameworkURL("https://hostName/PSIGW/
PeopleSoftListeningConnector");
```

You can also route your message to a specific Target Connector by modifying the request's ConnectorInfo object as follows:

```
 …
IBRequest ibRequest = new IBRequest();
…

// Send a message through the HttpTargetConnector for example.
ConnectorInfo connectorInfo = ibRequest.getConnectorInfo();

connectorInfo.setConnectorClassName("HttpTargetConnector");
connectorInfo.setField("URL","http://www.externalsite.com");
connectorInfo.setField("Method","POST");
…
```

You can control the routing of messages to PeopleSoft 8.4 systems by setting the destination node for messages and configuring the IntegrationGateway.properties file. For example, say you currently have PeopleSoft HR, Financials and CRM systems installed and you have built a custom listening connector to listen to events on an SAP MRP system. Assume that you have access to a method called getDestinationSystem( )on your listening connector that lets you know the destination of each message. If you were to call getDestinationSystem(), you would set the IntegrationGateway.properties file and listening connector as follows to accomplish your goals:

Set the application server settings in the IntegrationGateway.properties file as follows:

```
…
//Default PeopleSoft System
ig.isc.serverURL=//HRSERVER:9000
ig.isc.userid=HRUSERID
ig.isc.password=HRPASSWORD
ig.isc.toolsRel=8.40

#
## JOLT connect string settings for Application Server(s) with known
NODENAMEs.
#
ig.isc.FINANCIALS.serverURL=//FINSERVER:9000
ig.isc.FINANCIALS.userid=FINUSERID
ig.isc.FINANCIALS.password=FINPASSWORD
ig.isc.FINANCIALS.toolsRel=8.40

ig.isc.CRM.serverURL=//CRMSERVER:9000
ig.isc.CRM.userid=CRMUSERID
ig.isc.CRM.password=CRMPASSWORD
ig.isc.CRM.toolsRel=8.40
```

Make the following modifications to your listening connector:

```
…
IBRequest ibRequest = new IBRequest();
IbRequest.setMessageName("RoutingTest");
IbRequest.setRequestingNode("SourceSystem");
IbRequest.setPassword("myPassword");
…

// Get the name of the DestinationSystem from your proprietary //method. This
method would return one of the following ("HR", //"FINANCIALS", "CRM").
String destinationSystem = getDestinationSystem();
ibRequest.setDestinationNode(destinationSystem);
…
// Create a GatewayManager instance.
GatewayManager gm = new GatewayManager();

// Invoke the Connector Framework Manager.
ibResponse = gm.connect(ibRequest);
…
```

Building Error Handling and Logging into Listening Connectors

This section provides sample code for building error handling and logging into listenining connectors.

```
package com.peoplesoft.pt.integrationgateway.listeningconnector;

import ...

public class HttpListeningConnector extends HttpServlet {
            public void doGet(HttpServletRequest req,
            HttpServletResponse resp) throws ServletException, IOException {
            }

            public void doPost(HttpServletRequest req,
            HttpServletResponse resp) throws ServletException, IOException {

                  String actualResponse ="";
                  IBRequest request = null;
                  IBResponse response = null;

            try {

                  String inputString = MiscUtil.readerToString(new
                  InputStreamReader(req.getInputStream()));

                  // Log the actual Input String
                  Logger.logMessage("HttpListeningConnector: HTTP
                  Request", inputString, Logger.STANDARD_INFORMATION);

                  HttpListeningConnectorUtility util = new
                  HttpListeningConnectorUtility();
                  request = util.createIBRequest("XML", req, inputString);

                  // Use the GatewayManager to invoke the Integration
                  // Server and return its response.
                  GatewayManager conMgr = new GatewayManager();
                  response = conMgr.connect(request);

                  // Need to get the actual response from the
                  //IBResponse
                  actualResponse = response.getContentSectionAt(0);

      } catch (InvalidMessageException ime) {
                  ime.printStackTrace();
                  actualResponse = getErrorXml(ime);
                  Logger.logError("HTTPListeningConnector:
                  InvalidMessageException", request, response,
                  Logger.STANDARD_GATEWAY_EXCEPTION, ime);

      } catch (ExternalSystemContactException esce) {
```

```
                esce.printStackTrace();
                actualResponse = getErrorXml(esce);
                Logger.logError("HTTPListeningConnector:
                ExternalSystemContactException", request, response,
                Logger.STANDARD_GATEWAY_EXCEPTION, esce);

} catch (ExternalApplicationException esee) {
        esee.printStackTrace();
        actualResponse = getErrorXml(esee);
        Logger.logError("HTTPListeningConnector:
        ExternalApplicationException", request, response,
        Logger.STANDARD_GATEWAY_EXCEPTION, esee);

} catch (MessageMarshallingException mme) {
        mme.printStackTrace();
        actualResponse = getErrorXml(mme);
        Logger.logError("HTTPListeningConnector:
        MessageMarshallingException", request, response,
        Logger.STANDARD_GATEWAY_EXCEPTION, mme);

} catch (MessageUnmarshallingException mue) {
        mue.printStackTrace();
        actualResponse = getErrorXml(mue);
        Logger.logError("HTTPListeningConnector:
        MessageUnmarshallingException", request, response,
        Logger.STANDARD_GATEWAY_EXCEPTION, mue);

} catch (GeneralFrameworkException gfe)  {
        gfe.printStackTrace();
        actualResponse = getErrorXml(gfe);
        Logger.logError("HTTPListeningConnector:
        GeneralFrameworkException", request, response,
        Logger.STANDARD_GATEWAY_EXCEPTION, gfe);
}

// Return the message to the original requestor that
//invoked the Servlet
HttpListeningConnectorUtility.
                sendResponseBackToRequestor(actualResponse, resp);

// Log the actual output String
Logger.logMessage("HttpListeningConnector:
        HTTP Response", actualResponse, Logger.STANDARD_INFORMATION);

}       // end doPost()
```

```
}
```

Installing Connector Classes

You install connector classes on the local web server.

Target Connector Classes

To install a target connector class, copy the class to the following location on the local Web server from your Java Classes directory.

```
classes\com\peoplesoft\pt\integrationgateway\targetconnector
```

Listening Connector Classes

To install a listening connector class, copy the class to the following location on the local Web server:

```
classes\com\peoplesoft\pt\integrationgateway\listeningconnector
```

Registering Connectors

Before you can use a target connector you must register it on the Integration Engine. To register a connector, you should first load the connector information in the Gateways component using the Load button. Loading the connector makes its capabilities known to the Integration Broker.

Then you must assign the connector to the intended node from the Connector tab of the Node Definition component. Enter the connector ID that corresponds to your new connector and proceed to edit the properties as needed.

See Also

"Administering Basic Relationships," Registering Installed Target Connectors

Connector Templates

This section contains the following generic connector templates you can use as a starting point for connector development:

- Target Connector Template
- Listening Connector Template (Servlet Style)
- Listening Connector Template (Non-Servlet)

Target Connector Template

```
package com.peoplesoft.pt.integrationgateway.targetconnector;
import com.peoplesoft.pt.integrationgateway.common;

class  MyTargetConnector implements TargetConnector Interface  {

     public IBResponse send(IBRequest request) throws
                                     GeneralFrameworkException
                                     DuplicateMessageException
                                     InvalidMessageException
                                     ExternalSystemContactException
                                     ExternalApplicationException
                                     MessageMarshallingException
                                     MessageUnmarshallingException  {

          //Retrieve information from the Integration Broker
          //Request.
          String requestString = request.getContentSectionAt(0);

          //Retrieve Information about how the document is sent.
          ConnectorInfo ci = request.getConnectorInfo();

          //Retrieve Connector specific fields (URL, user, password
          //for example).
          String xxx = ci.getFieldValue("xxx");
          ...

          // Send document to its destination returning a
          //responseString.
          ...{code to interact with external system goes here}

          // Use the response to populate the ISResponse object
          IBResponse resp = new IBResponse()
          resp.addContentSection(null,responseString);
          return resp;
     }

     public IBResponse ping(IBRequest request)  throws
                                     GeneralFrameworkException
                                     DuplicateMessageException
                                     InvalidMessageException
                                     ExternalSystemContactException
                                     ExternalApplicationException
                                     MessageMarshallingException
                                     MessageUnmarshallingException   {
```

```
            //Retrieve Information about how the document is sent.
            ConnectorInfo ci = request.getConnectorInfo();

            //Retrieve Connector specific fields (URL, user, password
            //for example).
            String xxx = ci.getFieldValue("xxx");
            ...

            // Send a simple document to its destination just to see if
            //the destination is up.
            ...{code to interact with external system goes here. Throw
                  exceptions as needed.}

            // Return an empty IBResponse object
            return new IBResponse();
      }

      public ConnectorDataCollection introspectConnector() {
      ConnectorDataCollection conCollection = new
      ConnectorDataCollection();

            // Set the name of the connector.
            ConnectorData conData = new ConnectorData("MyTC");

            // Define the supported parameters for this Connector.
            conData.addConnectorField("xxx",true,"","");
            ...

            conCollection.addConnectorData(conData);

            return conCollection;
      }
}
```

Listening Connector (Servlet Style)

```
package com.peoplesoft.pt.integrationgateway.listeningconnector;
import com.peoplesoft.pt.integrationgateway.common;

//This is an example of using a Servlet when receiving the //External Request.
public class MyListeningConnector extends HttpServlet {

      public void service (HttpServletRequest request,
      HttpServletResponse response)
                        throws ServletException, IOException {
```

```
        //Developer retrieves the request and gets key information
        //here (such as UserName, password, messageName and
        //messageContent)
        ...

        //Developer creates the IBRequest object
        IBRequest request = new IBRequest();

        //Required information for an Integration Broker Request.
        request.setRequestingNode(UserName);
        request.setPassword(password);
        request.setMessageName(integrationService);
        request.addContentSection(null,messageContent);

        // Keep populating the IBRequest as needed (other set
        //methods are available…)

        GatewayManager gatewayManager = new GatewayManager();
        try {
               //Send the request into the Integration Broker.
               IBResponse response = gatewayManager.connect(request);

               //Hand back your response to the requestor.
               out.println(response.getContentSectionAt(0));
        } catch(MashallingException me) {

               // Handle Marshalling errors here. For example :
               out.println("<?xml version=\"1.0\"?>"+
                            "<MyResponse>"+
                                   "<Status>"+
                                   "<Code>1001</Code>"+
                                   "<Description>
                                   MarshallingException Occurred
                                   </Description>"+
                                   "</Status>"+
                            "</MyResponse>");
        } catch(UnmarshallingException ume) {
               // Handle UnmarshallingException  here.
        } catch(… ) {
               // Handle all other Integration Broker Exceptions.
        } finally  {
               //Cleanup work here… For example :
               out.close();
        }
}
```

Listening Connector (Non-Servlet)

```
package com.peoplesoft.pt.integrationgateway.listeningconnector;
import com.peoplesoft.pt.integrationgateway.common;

//This is an example of a regular class used to receive the //External
Request.
public class MyListeningConnector {

     //Somehow the external system makes a method call to this
     //method. Of course the Class and Method Name is entirely up
     //to the developer.

     public String callMyListeningConnector (String Request) {

     //Developer retrieves the request and gets key information
     //here (such as UserName, password, messageName and
     //messageContent)

     //Developer creates the IBRequest object
     IBRequest request = new IBRequest();

     //Required information for an Integration Broker Request.
     request.setRequestingNode(UserName);
     request.setPassword(password);
     request.setMessageName(integrationService);
     request.addContentSection(null,messageContent);

     //Keep populating the IBRequest as needed (other set
     //methods are available…)

     GatewayManager gatewayManager = new GatewayManager();
     try {
     //Send the request into the Integration Broker.
     IBResponse response = gatewayManager.connect(request);

     //Hand back your response to the requestor.
     return response.getContentSectionAt(0);
     } catch(MashallingException me) {

            // Handle Marshalling errors here. For example :
            return ("<?xml version=\"1.0\"?>"+
                   "<MyResponse>"+
                          "<Status>"+
                                 "<Code>1001</Code>"+
                                 "<Description>
                                 MarshallingException Occurred
```

```
                                    </Description>"+
                                "</Status>"+
                            "</MyResponse>");
        } catch(UnmarshallingException ume) {

              // Handle UnmarshallingException  here.
        } catch(… ) {

              // Handle all other Integration Broker Exceptions.
        } finally  {
              //Cleanup work here (if any)…
        }
  }
```

Using the Java XML DOM Wrapper for Connector Development

PeopleSoft provides an XML DOM Java Wrapper that enables you to manipulate message objects with PeopleCode instead of just a standard XML parser.

The section discusses how to:

- Work with the Java XML DOM wrapper.
- Use Java XML DOM classes.

Working with the Java XML DOM Wrapper

The Java XML DOM Wrapper is an abstraction layer over XML parsers that enables you to interpret, create or modify XML messages before you send them into or out of the Integration Gateway. The XML DOM wrapper provides an API equivalent to the PeopleCode API to support composing and transforming XML documents.

One very simple use case for the Java XML DOM wrapper is if you need to send incoming XML messages to the Integration Broker over HTTP, and that you need to read the content of the messages as they come in. Rather than parse each message on a character-by-character basis or use a parser, you can use the Java XML DOM wrapper to read the XML messages as they come into the Integration Gateway, add any information to them as necessary, and pass the messages on to the Integration Engine.

Another situation where you can use the Java XML DOM wrapper is when you need to send XML messages, but need to refine the messages before they get sent, such as change tag names, verify information in non-XML format, and so forth.

The Java XML DOM Wrapper:

- Enables you to navigate XML documents using methods such as GetParentNode, GetChild, getSibling, and so forth.

- Enables you to control the ordering of elements within XML documents.
- Supports additional XML features, such as namespaces, processing instructions and more.

Using the Java XML DOM Wrapper Classes

The Java XML DOM wrapper classes are shown in the table. The objects in these classes provide the basic methods for accessing and modifying DOM objects.

XmlDocument	Container for DOM that supports serialization and deserialization. Provides equivalent methods for dealing with XML documents as PeopleCode on the application server side.
XmlNode	Provides equivalent methods dealing with XML nodes as PeopleCode on application server side.
XmlNodeList	Provides equivalent methods dealing with XMLNodeList as PeopleCode on the application server side.
XmlException	Used to report any errors that occur while handling the Java XML DOM wrapper classes. Thrown when an XML DOMException is caught in DOM level. Most of XmlDocument/XmlNode methods throw the XmlException.

The Integration Broker Connector API documentation provides signature information for these classes.

See Also

PeopleTools 8.4 PeopleCode Reference PeopleBook, "XmlDoc Class"

Java XML DOM Code Example

```
public String simulateSendingMessage(String message) throws

     GeneralFrameworkException,

     DuplicateMessageException,
                                                    InvalidMessageException,

     ExternalSystemContactException,

     ExternalApplicationException,

     MessageMarshallingException,
```

```
     MessageUnmarshallingException,
                                                    XmlException {

//Create an XmlDocument object.

//Instantiate a XmlDocument object first. This step is mandatory.
XmlDocument xmlDoc = new XmlDocument();
XmlNode rootNode = null;

//Parse the string into the XmlDocument object.
try {

     //Fill in the XML structure/data with the real XML string
     xmlDoc.ParseXmlFromString(message);

     //Get the root XmlNode
     rootNode = xmlDoc.GetDocumentElement();
     } catch (XmlException xe) {
     throw new InvalidMessageException
     ("ExampleTargetConnector:InvalidMessageException",xe);
}

//Gather credentials from the Xml Document.
XmlNode itemNode;
XmlNode tmpNode;
float price;
float totalPrice = 0;

//Read all Message Items and calculate the totalPrice.
//Get the count of root XmlNode's immediate child XmlNode
for(int i=0; i < rootNode.GetChildNodeCount(); i++) {

//Get the number i child XmlNode
itemNode = rootNode.GetChildNode(i);

// Only process the Item nodes (any other tag will not be //processed).

//Get current XmlNode name
if (!itemNode.GetNodeName().equals("Item")) {
     continue;
}

if (itemNode == null) {
     String[] parms = {"Item"};
```

```
        throw new
        InvalidMessageException("ExampleTargetConnector:InvalidMessageException",
new     MessageCatalogEntry(10503,parms),null);
}

tmpNode = itemNode.FindNode("Price");

        price = Float.parseFloat(tmpNode.GetNodeValue());

        totalPrice += price * quantity;
}

return "<?xml version=\"1.0\"?>"+
        "<ExampleResponse>"+
        "<Status>"+
                "<Code>0</Code>"+
        "</Status>"+
        "<ResponseBody>"+
                "<TotalPrice>"+totalPrice+"</TotalPrice>"+
        "</ResponseBody>"+
        "</ExampleResponse>";
}
```

Understanding Developing and Implementing Connectors in the C/C++ Environment

PeopleSoft recommends that developers use Java for connector development. However, PeopleSoft also provides an environment for developing connectors using C/C++ through the Java Native Interface (JNI).

This section describes how to build target connectors for third-party systems developed using C++ and discusses:

- The development process to create connectors for the C/C++ environment.
- The steps to creating target connectors for the C/C++ environment.

Development Process

To develop connectors for the C/C++ environment you typically use Xalan and Xerces by Apache, Inc.

- Xalan is an XSLT processor that transforms XML documents into HTML, text, or other XML document types. Xalan uses two jar files: xalan.jar and xerces.jar.
- Xerces is a Java parser to parse XML.

For more information about these products and to download the Xalan jar files, see http://www.apache.org.

If you want to develop connectors and not use Xalan and Xerces, you can call a PeopleSoft-delivered Java connector, copy what you need from the message, and pass the information to the C/C++ environment.

Developing and implementing connectors for C/C++ environments is a process that involves the following steps:

1. Creating a Java target connector class.
2. Creating a JNI header file.
3. Implementing JNI header functions.
4. Building a DLL for the native library.
5. Adding the DLL to the system variables or adding it to the Web server path.

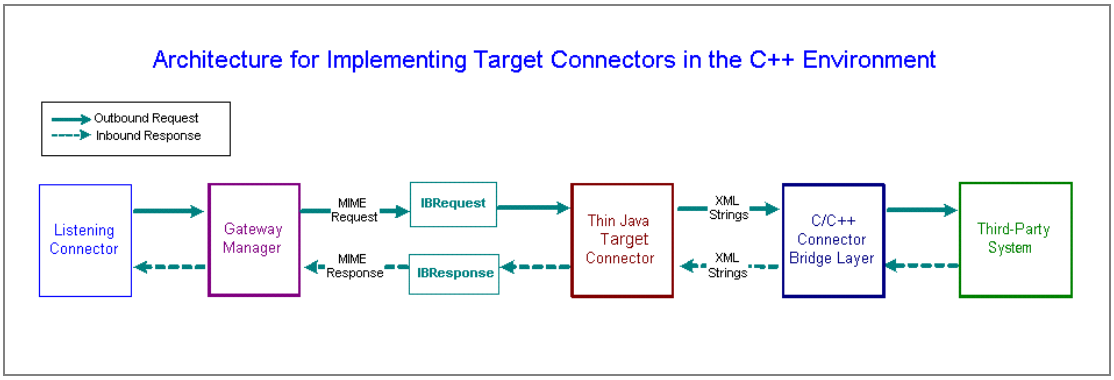
The information in this section provides an overview of these steps.

Creating a Java Target Connector

Building and implementing a connector in the C/C++ environment includes building a thin Java target connector to enable the Gateway Manager to load the connector the same way that it loads Java connectors, by parsing the IBRequest object.

The thin Java connector is also a gateway to the C/C++ connector. It is responsible for loading the connector and passing the XML string with business logic to one or more native (C/C++) methods.

The following diagram depicts the architecture for implementing target connectors in the C/C++ environment for communication with the Integration Broker Gateway.



Architecture for implementing target connectors in the C++ environment

The XML string contains the body from the MIME request string.

The IBResponse object is completely transparent to the native connector. All the information that the native side needs goes through the XML string or by strings that come from IBRequest.

PeopleSoft provides a template you can use as a starting point for developing the connector. In most cases you do not need to make any additions to the code since IBRequest provides the IBInfo and content body XML strings. However you can modify the IBInfo and content body XML strings in the C/C++ connector library, and declare additional native methods as necessary in the **psnativeconnector** section of the template.

Creating a JNI Header

After you create a Java connector, you must create a JNI header file, using the **javah** command. The javah command creates a C/C++ declaration. The JNI header file serves as a bridge between native methods you call within the Java target connector and those in the C++ environment.

Implementing the JNI Header Functions

When you implement the JNI Header functions, you pass the IBInfo using a native C++ functional call from the Java environment to the third-party system. In doing so you pass the business logic to the C++ system.

Building a DLL for the Native Library

To build a DLL for the native library, compile the C++ functions generated by the previous steps.

Registering the DLL

Register the DLL you built for the native library by adding it to the system variables or adding it to the Web server path.

Creating Target Connectors for the C/C++ Environment

This section provides step by step instructions for creating a target connector for the C/C++ environment.

Creating Target Connector for the C/C++ Environment

To create a target connector for the C/C++ environment:

1. Create a Java target connector.

    a. Copy the code from the following Java target connector template into a text editor:

```
public class CPlusPlusTargetConnector implements TargetConnector {
              // Native method Declaration
              public native String native_simple(String IBReqString,
                     String[] xmlStringArray);

              public IBResponse send(IBRequest request)  {
```

```
                IBResponse response = null;

                try {
                        String ibReqString = request.getIBInfoString();
                        String requestXml1 = request.getContentSectionAt(0);
                        String requestXml2 = request.getContentSectionAt(1);
                        …

                        // Assign to String[] xmlStringArray
                        // Load native lib that implements the connector
                        System.loadLibrary("psnativeconnector");

                        responseStr = native_simple(ibReqString,
                        xmlStringArray);

                        response = new IBResponse();
                        response.addContentSection(null, responseStr);
        }
        catch (Exception ioe) {
                        throw new
GeneralFrameworkException(ioe.getMessage());
        }
        return response;
}
```

b. Compile the code.

Using a Java compiler, compile the CPlusPlusTargetConnector.java file into a CPlusPlusTargetConnector.class file. Then copy the class file into the target connector directory on the Web server. The CLASSPATH environment variable should point to the Integration Broker.jar file.

c. Install the connector.

d. Register the connector.

2. Create a JNI header file.

At a DOS prompt or UNIX shell command prompt, enter the command and press **Enter**:

```
hisdir>javah –jni
com.peoplesoft.peoplesoft.pt.integrationgateway.targetconnector.CplusPlusTarg
etConnector
```

3. Implement the JNI header file.

a. Copy the method declaration output from the previous step to a C++ file. The following code shows sample method declaration output from the **javah** command.

```
JNIEXPORT jstring JNICALL
Java_com_peoplesoft_pt_integrationgateway_targetconnector_CPlusPlusTargetC
onnector_native_1simple (JNIEnv *env, jobject obj, jstring ibInfo,
jobjectArray contentArr):
```

b. Convert the **jstring** to an ANSII string or Unicode string.

To convert the jstring to Unicode, follow this example:

```
const TCHAR * tStr;

tStr = env->GetStringChars(jstrXml, NULL);
```

To convert the jstring to ANSII, follow this example:

```
Const char* string;

string = env->GetStringCharsUTF(jstrXml, NULL);
```

You may now parse the XML as necessary.

4. Build the PSNativeConnector DLL.

a. Compile the C++ functions from the previous step into a DLL file.

b. Save the file. You do not have to use **PSNativeConnector.DLL** as the name for the file, however the name you use *must* match the connector name in the connector class file. If you use another name for the DLL, enter the new name for the connector in the following line of the connector class file:

```
System.loadLibrary("psnativeconnector");
```

5. Register the DLL.

Register the DLL by adding it to the system variables or adding it to the path of the Web server environment.

To add the DLL path to the system variables, select **Control Panel**, **System**, **Environment**.

To add the DLL path to the Web server environment, follow the instructions provided for the Web server you are using.

WebLogic Web Server

a. Open a command prompt or shell command.

b. Append the path to the library at the end of the line:

```
:runWebLogic
echo on
set PATH=.\bin;%PATH%;%PATH_TO_YOUR_NATIVE_LIBRARY%
```

WebSphere Web Server

a. Open a command prompt or shell command.

b. Type the following and press ENTER:

```
C:\Apps\WebSphere\AppServer\bin\startServer.bat
```

c. Append the path to the library in this line:

```
SET PATH=;%PATH_TO_YOUR_NATIVE_LIBRARY%;%WAS_HOME%/
bin;%JAVA_HOME%/bin;%JAVA_HOME%/jre/bin;%PATH%
```

See Also

Installing Connector Classes

Registering Connectors

"Administering Basic Relationships" Registering Installed Target Connectors

Reusing Connector Code

When you create a custom connector, you can reuse code to leverage functionality of an existing connector through inheritance or delegation.

This section discusses how to:

- Reuse connector code through inheritance.
- Reuse connector code through delegation.

Reusing Connector Code through Inheritance

Using inheritance you can extend an existing connector and provide additional behavior.

```
public Class MyHttpTargetConnector   extends HttpTargetConnector {
public IBResponse send(IBRequest request) throws
                                              GeneralFrameworkException,
                                              InvalidMessageException,
                                              ExternalSystemContactException,
                                              ExternalApplicationException,
                                              MessageMarshallingException,
                                              MessageUnmarshallingException,
                                              DuplicateMessageException  {

    // Do connector specific logic here possibly modifying the
    // request…
    IBResponse response = super.send(request);
```

```
        // Do connector specific logic here possibly modifying the
        // response…
        }
}
```

Reusing Connector Code through Delegation

Using delegation, you can simply reuse the code from an existing connector to create a new customized connector, as shown in the example.

```
public Class MyHttpTargetConnector   implements TargetConnector {
     public IBResponse ping(IBRequest request) throws
                                               GeneralFrameworkException,
                                               InvalidMessageException,
                                               ExternalSystemContactException,
                                               ExternalApplicationException,
                                               MessageMarshallingException,
                                               MessageUnmarshallingException,
                                               DuplicateMessageException  {

     HttpTargetConnector httpTargetConnector = new
            HttpTargetConnector();
     return httpTargetConnector.ping(request);
}
public IBResponse send(IBRequest request) throws
                                               GeneralFrameworkException,
                                               InvalidMessageException,
                                               ExternalSystemContactException,
                                               ExternalApplicationException,
                                               MessageMarshallingException,
                                               MessageUnmarshallingException,
                                               DuplicateMessageException  {

     // Do connector specific logic here possibly modifying the
     // request…
     HttpTargetConnector httpTargetConnector = new
            HttpTargetConnector();
     IBResponse response = httpTargetConnector.send(request);

     // Do connector specific logic here possibly modifying the
     // response…
}

public ConnectorDataCollection introspectConnector() {
     HttpTargetConnector httpTargetConnector = new
```

```
            HttpTargetConnector();
        ConnectorDataCollection cdc =
            httpTargetConnector.introspectConnector();

        // Possibly add specific fields here…
        Return cdc;
        }
    }
```

Testing Message Processing Using Send Master

Send Master is a utility that enables you to test overall message processing in the Integration Broker, including listening connector functionality, target connector functionality, connector introspection and Integration Broker transactions.

Send Master enables you to Post any data format, including the PeopleSoft MIME message format, to web and application servers over HTTP(s).

You can also use Send Master to simultaneously test groups of different types of messages, as well as stress test your system.

This section discusses how to:

- Open Send Master.
- Create projects.
- Test projects.
- Create groups of projects.
- Manage groups of projects.
- Test groups of projects.
- Share projects and groups of projects.
- View test output.

Understanding Send Master

Send Master is a utility you can use to send messages over HTTP(S). Send Master enables you to:

- Construct messages and send them over HTTP(S) to Web servers or PeopleSoft application servers to test target connector and listening connector functionality on the Integration Gateway.
- Test connector introspection

- Perform GET functions and ping application messaging gateways and third-party servers.
- Mimic sending messages to the Integration Broker, as well as messages you send from the Integration Engine to the Integration Gateway.
- Automate the testing process by enabling you to create groups of different types of messages and test them with a click of a button.
- Stress test groups containing projects with a click of a button.

Opening Send Master

To access Send Master, you must have the Integration Gateway installed. To open Send Master you launch the startsendmaster.bat file (Windows) or startsendmaster.sh file (UNIX) located in the Integration Gateway directory.

The location of the file depends on the web server you are using:

- WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW

- WebSphere:

c:\websphere\AppServer\installedApps\peoplesoft\PSIGW

Opening Send Master

To open Send Master

1. Use the following instructions based on your system environment::

   Windows

   a. Open Windows Explorer and navigate to the Integration Gateway directory.

   b. Double-click startsendmaster.bat.

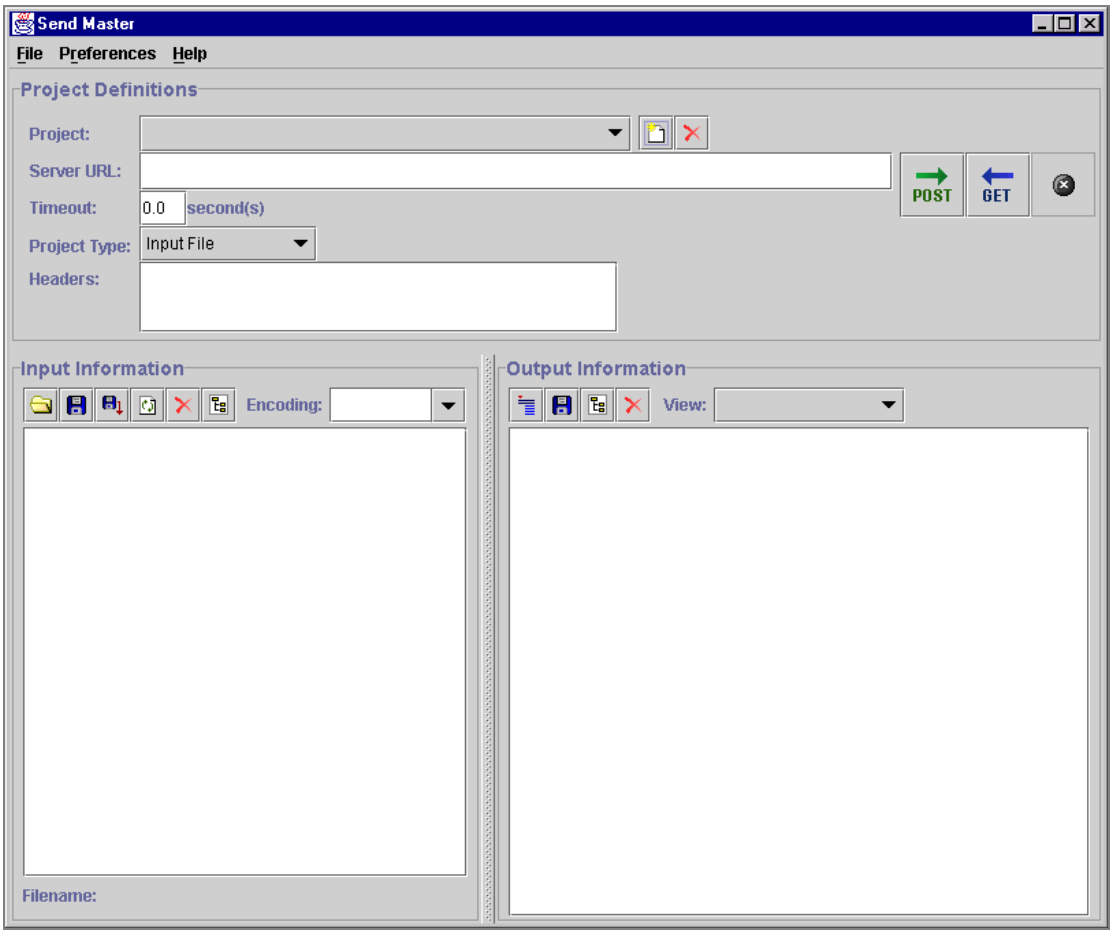
   UNIX

   a. Open a shell command prompt and navigate to the Integration Gateway directory.

   b. Enter the following:

   ```
   /startsendmaster.sh
   ```

   c. Press **Enter**.



Send Master Workspace. No fields or buttons are enabled until you define or select a Project.

Understanding Send Master Workspaces

The Send Master workspace features three main sections as well as several menus and toolbars with which you can work.

Dropdown Menus

Send Master features three dropdown menus.

Menu	*Menu Option*	*Shortcut*	*Action*
File	Automation	Alt + A	Opens the Automation Workspace.
	Exit	Ctrl + E	Closes Send Master.
Preferences	Output word wrapping	None	Turns word wrapping on or off in the Output Information section. When checked, the option is on.

Menu	*Menu Option*	*Shortcut*	*Action*
	Input word wrapping	None	Turns word wrapping on or off in the Input Information section. When checked, the option is on.
Help	About Send Master…	None	Displays Send Master version information.

Project Workspace

When you open Send Master, the Project Workspace displays. You use the Project workspace to define, modify and test a Send Master project. A Send Master project is a collection of message components, values and parameters that defines the message you want to test and how you want to test it.

The Send Master Project workspace features three sections:

Project Definitions — Area where you add and define a new Send Master project. The information you specify in this section includes the Web server URL where you POST or GET messages, specify the project type and more.

Input Information — Area where you create and format MIME messages.

Enables you to read in files using the any of the following character sets: ASCII (default), ISO859_1, UTF8 and UTF-16. You can also add other character sets as needed.

You need to know the message format that the connectors, application servers, and so forth are expecting and incorporate the appropriate tags and components into the message body. For example, to communicate with PeopleSoft systems, you must specify the message name and requesting node.

When you work with an Integration Broker project type, this section displays tabs where you can specify input files, destination nodes, and more.

Output Information — Displays information returned when you perform a GET or POST on a Web server.

When you work with MIME messages, you can use the provided View dropdown list to choose to view the entire raw message response, message meta data, or individual sections of the response.

When you are working with message types other than MIME, you can view the raw message response only.

The Send Master Project workspace features three toolbars:

- Project Definitions toolbar.
- Input Information toolbar.

- Output Information toolbar.

The Project Definitions toolbar features two buttons.

Click the **Add a Project** button to creates a new project.

Click the **Delete the Selected Project** button to deletes the selected project and all data associated with it.

The Input Information toolbar appears only when you work with Input File project types and features six buttons.

Click the **Open File** button to open an existing file and display it in the Input Information area.

Click the **Save File** button to save the contents displayed in the Input Information area, using a filename and location that you specify.

Click the **Save File As** button to save the currently displayed file, using another name and/or location that you specify.

Click the **Refresh the Current File** button to reload and display the last saved version of the current file.

Click the **Remove File Reference** button to delete the contents of the Input Information area.

Click the **If Valid XML, Format** button to format the code displayed in the section to make it more readable. The formatting done includes indenting lines, and so forth. This button is valid only if the file displayed is an XML file.

The Output Information toolbar features four buttons.

Click the **View Header Information** button to display only the contents within the header tags of the selected message.

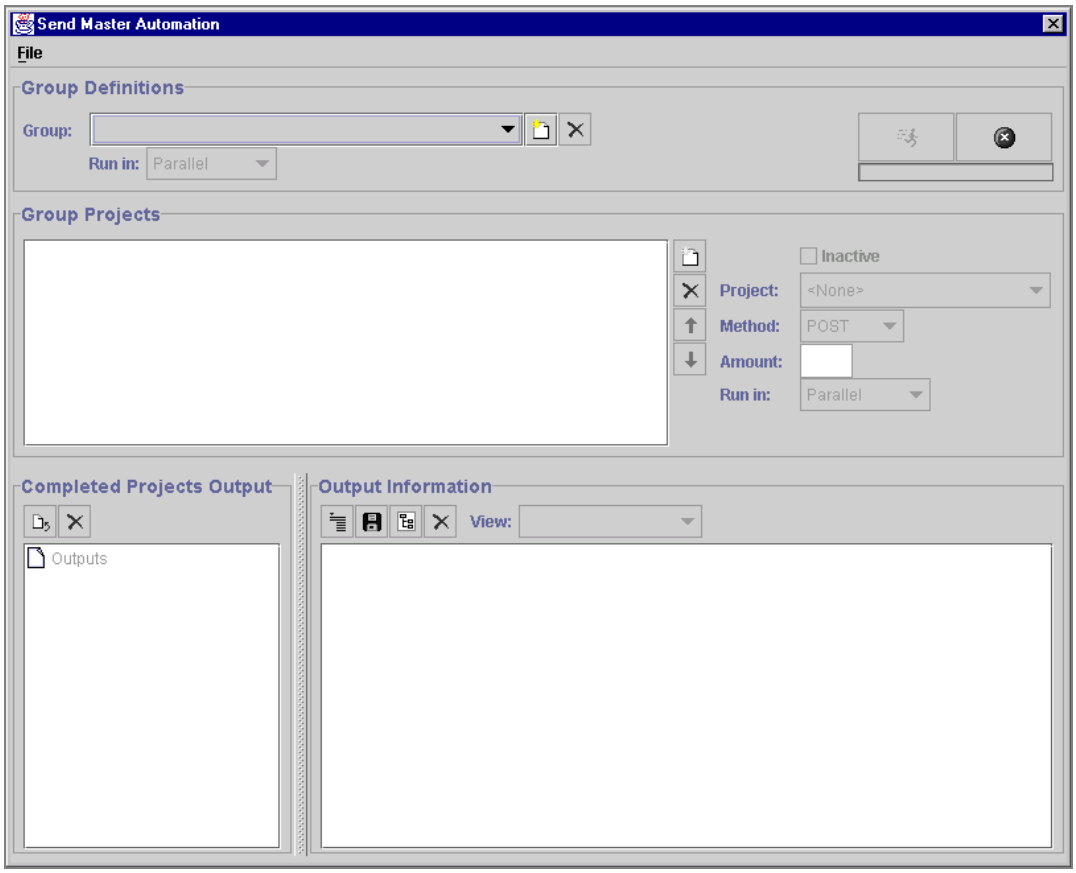
Click the **Save Output** button to save the information in the Output Information section using a filename and location that you specify.

Click the If Valid XML, Format button to format the code displayed in the section to make it more readable. The formatting done includes indenting lines, and so forth. This button is valid only if the file displayed is an XML file.

Click the **Clear Output** button to delete the contents of the Output Information area.

Automation Workspace

The Automation workspace enables you to test groups of projects, as well as stress test a project or group of projects. You can access the Automation workspace by opening Send Master and selecting **File**, **Automation**.



Send Master Automation workspace. No buttons or fields are active until you define or select a Group.

The Send Master Automation workspace features four sections:

Group Definitions	Create, select or delete a group of projects. Enables you to choose to run the projects in the group all at once, in sequence or at intervals you specify.
Group Projects	Add, remove and arrange projects in a group. For each project you add to a group you can select the Method to invoke, such as GET or POST. You can also specify the number of times to run each project, and specify to run project instances all at once, in sequence or at defined intervals.
Completed Projects Output	Provides processing information for each project in a group, including the number of project instances processed, total time to process all project instances, the

	average amount of time to process a project instance, and more.
Output Information	Displays information returned when you perform a GET on a Web server. When you work with MIME messages, you can use the provided View dropdown list to choose to view the entire raw message response, message meta data, or individual sections of the response. When you are working with message types other than MIME, you can view the raw message response only.

Automation Workspace Toolbars

The Send Master Automation workspace features three toolbars:

- Group Definitions toolbar.
- Output Information toolbar.
- Completed Project toolbar.

The Group Definitions toolbar features two buttons.

	Click the **Add a Group** button to creates a new group.
	Click the **Delete the Selected Group** button to delete the selected group and all projects and data associated with it.

The Output Information toolbar features options to work with information in the Output Information section.

	Click the **View Header Information** button to display only the contents within the header tags of the selected message.
	Click the **Save Output** button to save the contents of the Output Information area, using a filename and location that you specify.
	Click the **If Valid XML, Format** button to format the code displayed in the section to make the contents more readable. The formatting includes indenting lines, and so forth. This button is valid only if the file displayed is an XML file.
	Click the **Clear Output** button to delete the contents of the Output Information area.

The Completed Project toolbar features options to work with information in the Completed Project section of the Automation workspace.

Click the **Export Results to File** button to display a text file that contains processing information about the completed project, such number of messages processed, the total time to process the messages, the average time to process a message, and so forth.

Click the Clear Results button to clear the contents currently displayed in the area.

Understanding Send Master Project Types

The following table summarizes the project type to use based on what type or message you want to test or tasks you want to perform.

Input File Project Type

The Input File project type enables you to test servers that are expecting XML data over HTTP(S).

Integration Broker Project Type

The Integration Broker project type also enables you to test servers that are expecting MIME data in PeopleSoft format over HTTP(S).

The following table describes the type of project to use based on the type of communication you want to test.

To test …	*… use this project type*
Using Integration Broker to send PeopleSoft 8.4 messages to: • Other PeopleSoft 8.4 systems (cross-node communication) • Remote PeopleSoft gateways	Integration Broker
Connector introspection.	Integration Broker
Communication to an Integration Gateway via connectors: • 3rd-party systems • PeopleSoft 8.1 systems	Input File or Integration Broker. The project type you use depends on the connector you are testing and the input the connector is expecting. Use an Input File project to communication to the Integration Gateway by a connector that is expecting XML (for example, the HTTP Listening or the 8.1 Listening Connector. Use an Integration Broker project to pass information to the PeopleSoft Listening Connector.

To test …	*… use this project type*
Communications from a PeopleSoft 8.1 system to a PeopleSoft 8.4 system.	Input File
Sending SOAP messages to a PeopleSoft 8.4 system.	Input File
Target connectors that sit on the Integration Gateway.	Integration Broker
Listening connectors that use HTTP(s) to communicate.	Input File or Integration Broker. The project type you use depends on the connector you are testing and the input the connector is expecting. For example: You would use an Input File project to communication to the Integration Gateway by a connector that is expecting XML (for example, the HTTP Listening or the 8.1 Listening Connector. You would use an Integration Broker project to pass information to the PeopleSoft Listening Connector.
Communications from a third-party system to Integration Broker.	Input File

Understanding Headers Used in Send Master

Send Master enables you to specify HTTP, PeopleSoft and connector headers. Use the following information as a guide for entering header information in Send Master.

Header Type	*Project Type*	*Location*	*Description*
HTTP header	• Input File • Integration Broker	Project Definition section, Headers box.	Use to provide HTTP protocol header information about the message at the server level and relates to how you are sending an entire message. Here you can specify cookies, content-type, encoding, sending program information, and so forth.
Input File header	Integration Broker	Input Information section, Input Files tab.	Use to provide information specific to each input file. Here you can specify message-specific information.

Header Type	*Project Type*	*Location*	*Description*
Connector header	Integration Broker	Connector Introspection window. Input Information section, Additional Headers tab, Edit Connector Info.	Use to provide required and optional headers that connectors need to pass information and process the message request. Here you can specify information such as send the message compressed, encoding information, and so forth. You can specify connector header information only while editing connector information in an Integration Broker project type.
PeopleSoft Header	Integration Broker	Input Information section, Header Information and Additional Header Information tabs.	MIME wrapper placed around input file messages. Contains the information required to route the message through the Integration Broker, including message name, message type and requesting node.

Creating Input File Projects

This section describes the steps for creating an Input File project in Send Master. This section discusses how to:

- Create an Input File project type.
- Add or create an input file for an Input File project.
- Export project settings.
- Post a project.

Creating an Input File Project Type

The following information describes how to create an Input File project type

To create an Input File project type

1. Click the New Project button to the right of the **Project** field and enter the name of the project.
2. Click **OK**.
3. Enter the **Server URL** of the server with which to communicate.
4. Enter a **Timeout** value.

   The **Timeout** value determines the amount of time Send Master attempts to process a message. If the request does not complete in the time specified, processing stops. The

default value is 0, meaning there is no timeout and Send Master will attempt to process project messages indefinitely. A typical timeout interval is 60 seconds.

5. In the **Project Type** dropdown list, select **Input File**.

6. In the **Headers** box, enter any pertinent HTTP header information for the message.

Adding or Creating an Input File for an Input File project

This section describes how to add or create an input file for the project.

To add or create an input file for an Input File project

1. In the **Input Information** section enter the message contents.

   You can compose the content of the message in the area provided, or import a file.

   To import a file, click the **Open File** button and select a file. The name of the imported file displays under the Input Information section.

2. Modify the message, if necessary.

3. To apply a different character set, select one from the **Encoding** dropdown list. By default, message content displays in ASCII format.

4. Click **Save**.

After you create an input file you can modify and format message content. Use the following tips when you work with input files for Input File project types:

- Use the **Refresh** button to revert to the last saved version of the input file.
- If the message content is XML, use the **Format** button to indent lines of code.
- Use the **Delete** button, to delete the contents of the section.

Posting the Input File to the Server

After you create the Input File project type, add or create and save the input file for the project, click Post to post the file to the server.

Any server response to the message you post displays in the **Output Information** section.

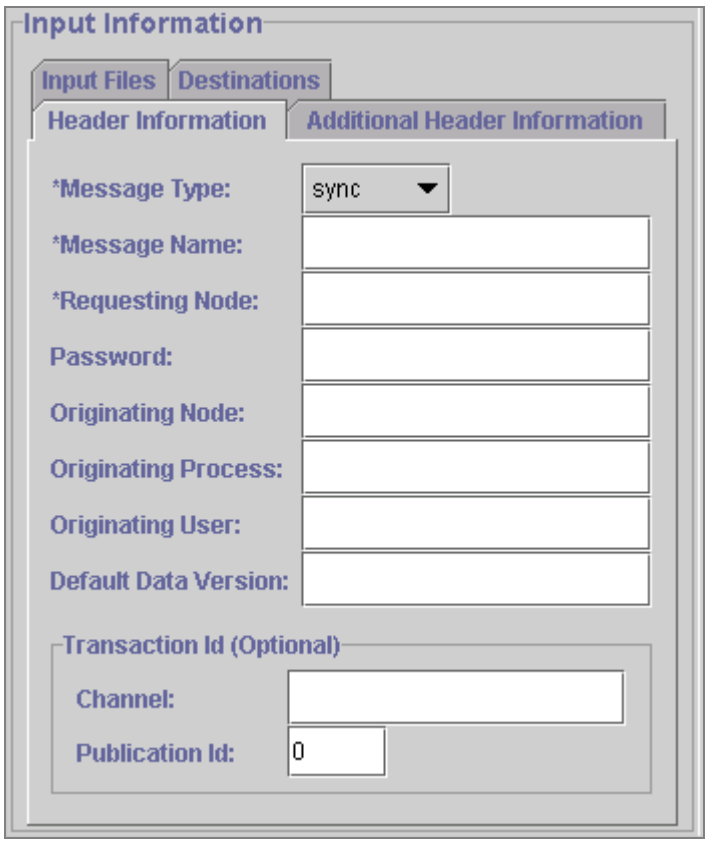
See Also

Understanding Send Master Project Types

Working with Integration Broker Projects

When you create a Send Master Integration Broker project, you supply Send Master with information to build the IBInfo section of an XML message. In addition, you specify connector information, add cookie information, specify destination nodes, and more.

When you work with this project type, the Input Information section differs from when you work with an Input File project. With an Integration Broker project type, the Input Information section displays a number of tabs that you use to define project parameters and work with messages.



Input Information Section for Integration Broker Project Types.

Header Information Tab

Use the Header Information tab to create PeopleSoft message headers. You can specify the message type, such as synchronous, asynchronous or ping. You also use this tab to enter the name of the message, the requesting node, source node and more. The tab features the following controls:

Message Type Identifies the message type. Values are:

- *Sync*: Specifies that the message you are testing is synchronous.

- *Async*: Specifies that the message you are testing is asynchronous.
- *Ping*: Test the application server to make sure it is available and accepting requests.

Message Name	Identifies the name of the message.
Requesting Node	Identifies the name of the node that is making the request.
Password	(Optional) Identifies the password as entered in the node definition, If password authentication is used. This entry allows password authentication to allow processing of the message.
Originating Node	(Optional) Identifies the name of the node that started the process.
Originating Process	(Optional) Identifies the name of the process where the publish originated. For example, a message published from the Inventory definitions page would have a process name of *INVENTORY DEFIN*.
Originating User	(Optional) Identifies the user ID login where the message was initially generated.
Default Data Version	(Optional) Identifies the default version to use for all messages contained in the MIME request.
Channel	(Optional) Identifies the name of the PeopleSoft channel expecting the message.
Publication ID	(Optional) Identifies the unique identifier for the message. Used to track the message through the PeopleSoft Integration Broker. The value increments by one each time you send the message through the system.

Input Files Tab

You use this tab to add, remove and order the sequence of input files. Apply non-repudiation, Base 64 encoding and compression. Also enables you to apply character sets, add version information and add header information. This tab features the following controls:

Input Files	(Optional) Area where you can create or import a message for testing.
Non-repudiation	(Optional) Identifies whether to apply non-repudiation to the message.
Base 64 Encode/ Compress	(Optional) Identifies whether to apply Base 64 encoding/compression to the message.
Encoding	(Optional) Identifies the character set to apply to the message.
Version	(Optional) Identifies the version number applies to the individual message. This value is an integer.

Headers	(Optional) Identifies input file headers to pass to the application server.

Destinations Tab

Use this tab to add or remove destination nodes. Set final destination node if sending information through a hub. The tab features the following controls:

Destination Nodes	(Optional) Identifies destination nodes for the message.
Final Destination	(Optional) Identifies the final destination node. Check to specify the selected node as the final destination. Use this option when working with a hub configuration.

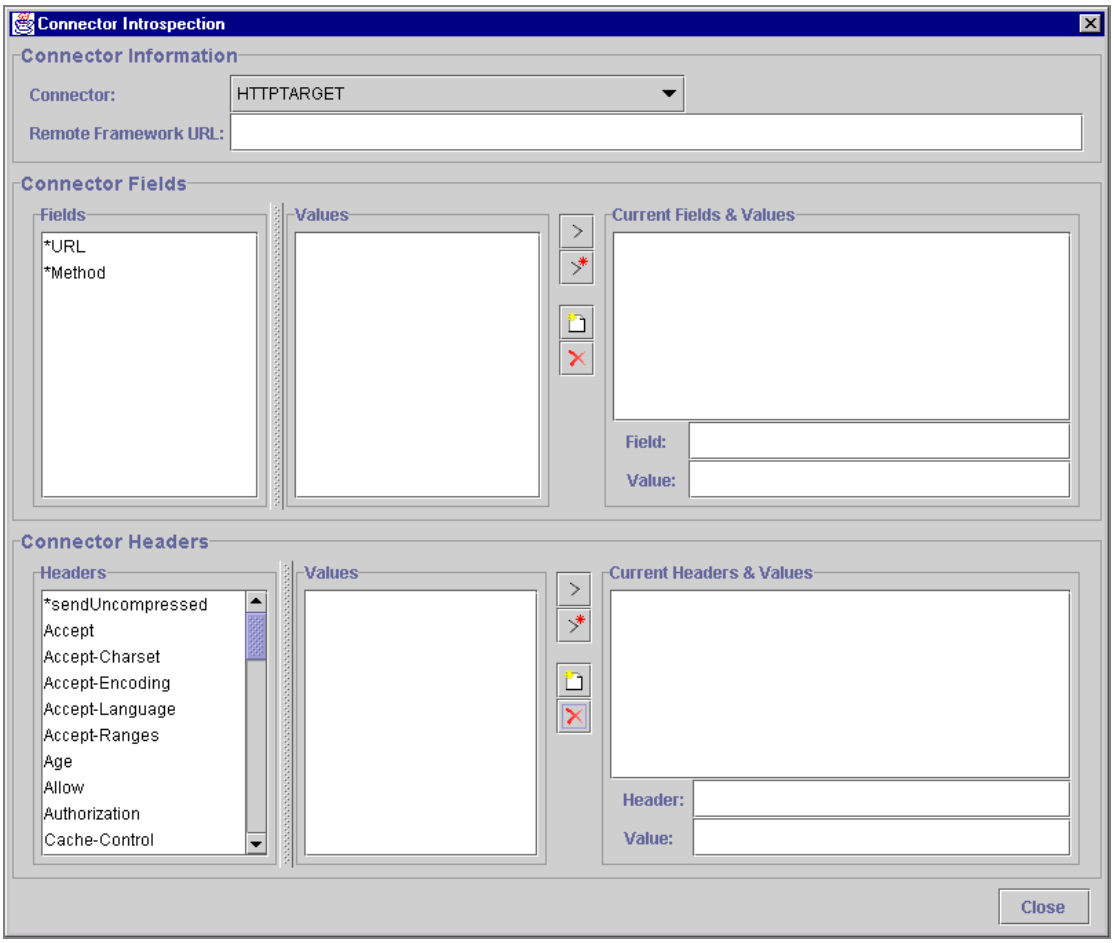
Additional Header Information Tab

Use the Additional Header Information tab to add sub channel, visited node and cookie information to the connector header. The tab features the following controls:

Sub Channel	(Optional) Identifies sub processes for the channel.
Visited Nodes	(Optional) Identifies nodes through which the message has passed. Separate by semi-colons. Enables you to mimic visited node information populated when sending PeopleSoft messages through the Integration Broker
Cookies	(Optional) Identifies cookies that the server may require.
Use Connector	(Optional) Identifies whether to use a connector for the message.
Edit Connector Info	Click the **Edit Connector Info** button to introspect the Integration Gateway. The Connector Introspection window displays and enables you to choose a connector, specify connector fields and values, and enter connector header information.

Connector Introspection Window

The Connector Introspection window enables you to work with all of the target connectors loaded on the Integration Gateway. Use this window to select a target connector, remote URL, connector fields and values (properties) and connector header information.



Connector Introspection window

Connector Information Section

The Connector Information section of the Connector Introspection window provides a dropdown list that enables you to select a target connector loaded on the Integration Gateway. It also enables you to specify a URL other than the server URL specified in the Project Definitions section on the main Send Master window.

Connector	Identifies the target connector to use for the project..
Remote Framework URL	Identifies the URL to which to send a message. This value overrides the server URL specified in the Project Definitions section.

Connector Fields Section

The Connector Fields section enables you to select specific target connector properties (in the form of field and value pairs) for the message instance, as well as create and add new properties. This section features the following fields:

Fields	Displays all required and optional connector properties for the target connector you select from the Connector dropdown list. An asterisk (*) denotes a required property.
Values	Displays all pre-defined connector property values for the corresponding properties in the Fields box. If a value for a property is not define,
Current Fields and Values	Shows a list of all selected properties and their values for the entire project.

The Connector Fields section also features controls that enable to add, remove and delete fields and values.

	Click the Add Selected Field and Value button to move the selected value in the Field column to the Current Field & Values column.
	Click the **Add All Required Fields and Default Values** button to add all required properties and their default values in the Field column to the Current Fields & Values column.
	Click the **Create a New Field** button to create a new connector property and enter a value for it..
	Click the Remove Field button to remove a connector property and its value from the Current Fields and Values list.

Connector Headers Section

The Connector Headers section enables you to specify required and optional headers that the selected connector needs to pass information and process the message request.

The fields and controls with which you can work in the Connector Introspection window are:

Fields	Displays all required and optional connector headers for the target connector you select from the Connector dropdown list. An asterisk (*) denotes a required header.
Values	Displays all pre-defined connector header values for the corresponding header in the Fields box.
Current Fields and Values	Shows a list of all selected connector headers and their values for the entire project.

The Connector Headers section also features controls that enable to add, remove and delete connector header fields and values.

	Click the **Add Selected Headers and Default Values** button to move the selected header and its value in the Headers column to the Current Headers & Values column.

Click the **Add All Required Headers and Default Values** button to add all required connector headers and their default values in the Current Headers & Values column.

Click the **Create a New Field** button to create a new connector header and enter a value for it.

Click the Remove Field button to remove a connector header and its value from the Current Headers & Values list.

Creating Integration Broker Projects

This section describes the steps for creating Integration Broker project types in Send Master. The section discusses how to:

- Create an Integration Broker project type.
- Add PeopleSoft Header information to the project.
- Add input files to the project.
- Specify destination nodes.
- Specify additional header information.
- Specify connector information.
- Post the project.

Creating Integration Project Types

To create an Integration Broker project type

1. Click the New Project button to the right of the Project field and enter the name of the project.
2. Click **OK**.
3. Enter the **Server URL** of the server with which to communicate.
4. Enter a **Timeout** value. The default is 0, meaning there is not timeout.

   The **Timeout** value determines the amount of time Send Master attempts to process a message. If the request does not complete in the time specified, processing stops. Usual timeout is about 60 seconds.
5. In the **Project Type** dropdown list, select **Integration Broker**.
6. In the **Headers** box, enter pertinent HTTP header information for the message.

Adding PeopleSoft Header Information to an Integration Broker Project

To add PeopleSoft header information to the project

1. In the **Input Information** section, click the **Header Information** tab, if not already selected:
2. Select or enter values for the following required fields:
    - Message Type
    - Message Name
    - Requesting Node
3. Enter values in any of the remaining optional fields as appropriate for your project.

Adding Input Files to an Integration Broker Project

To add input files to the project

1. Select the **Input Files** tab and add a file or files to the project.
2. Click the **Add Input File** button and select a file to add to the project.
3. Repeat Step 2 to add additional files to the project.
4. Enter values in any of the remaining optional fields as appropriate for your project.

Key points:

- To change the order of a file in the project, use the arrow buttons to move the file up or down in the list.
- To remove a file from the project, select the file and click the **Delete** button.

Specifying the Destination Nodes for an Integration Broker Project

To specify destination nodes

1. Select the **Destinations** tab.
2. Click the **Add Destination Node** button. Enter the name of the destination node and click **OK**. Repeat to enter additional destination nodes.
3. To specify a final destination, select a node in the **Destination Nodes** list, and check the **Final Destination** Box.

Specifying Additional Header Information for an Integration Broker Project

To specify additional header information.

1. Click the **Additional Header** Information tab.
2. Enter values in any of these optional fields.

Specifying Connector Information for an Integration Broker Project

To specify connector information

1. Click the Additional Header Information tab.
2. Check the **Use Connector** box, and click the **Edit Connector Info** button. The Connector Introspection window displays.
3. Select a **Connector** from the dropdown list.

   All required and optional connector properties and their predefined property values populate the **Fields** list in the **Connector Fields** section.

   All required and optional and connector headers and their predefined header values populate the **Fields** list in the **Connector Headers** section.

   An asterisk (*) denotes a required property or header.
4. In the Remote Framework URL, enter a URL if you would like to send the message to a location other than the server URL you entered in the **Project Definition** section.
5. Add connector properties to apply to the current project.

   - To add all required properties and their default values, in the Connector Fields section click the **Add All Required Fields and Default Values** button. Each required field and its default value display in the **Current Fields and Values** list, separated by a colon.
   - To specify a property value when there is no default value or no predefined values are provided, in the **Current Fields and Values** list click the property name. Note that the property name populates the **Field** field at the bottom of the section. In the **Value** field, enter a value for the property.
   - To add specific connector properties, in the **Connector Fields** section, click a property. Any predefined values for the property display in the **Values** list. Select the appropriate value and click the **Add Selected Field and Value** button. The property and value you select display in the **Current Fields and Values** list, separated by a colon.
   - To create a connector property and value, click the **Create a New Field** button and enter the property name in the **Field** field and the property value in the **Value** field.

- To remove a property and property value from the **Current Fields and Values** list, click the **Remove Field** button.

6. Add connector headers to apply to the current project.

   - To add all required headers and their default values, in the **Header Fields** section click the **Add All Required Headers and Default Values** button. Each required header and its default value display in the **Current Headers and Values** list, separated by a colon.

   - To specify a header value when there is no default value or no predefined values are provided, in the **Current Headers and Values** list click the header name. Note that the property name populates the **Header** field at the bottom of the section. In the **Value** field, enter a header value.

   - To add specific headers, in the **Header Fields** section click a header. Any predefined values for the header display in the **Values** list. Select the appropriate value and click the **Add Selected Header and Value** button. The header and value you select display in the **Current Headers and Values** list, separated by a colon.

   - To create a header and value, click the **Create a New Field** button and enter the header name in the **Field** and the header **Value**.

   - To remove a header and its value from the **Current Headers and Values** list, click the **Remove Field** button.

7. Click **Close**.

Posting an Integration Broker Project

To post an Integration Broker project

After you create the project click the **Post** button to post the project to the server.

When you POST a message using the Integration Broker project type, a MIME response message is returned. If you POST data to a PeopleSoft Listening Connector, the MIME response displays in the **Output Information** section of the Project Workspace.

See Also

Understanding Send Master Project Types

Working with Integration Broker Projects

Creating Groups of Projects

To create a group of projects

1. Select File, Automation.
2. In the Group Definitions section:
   a. Click the **Add a new group** button and enter a name for the group. Click **OK**.
   b. From the **Run In** dropdown list, select one of the following options. The options determine how the projects in the group run.

      Parallel. Run all projects in the group at the same time.

      Succession. Run projects in the group in succession, one after another.

      Time Lapse. Run projects in the group in the interval you specify in the **Delay** field.
3. Add projects to the group.
   a. In the **Group Projects** section, from the **Projects** dropdown list, select a project.
   b. Click the **Add a new project** button to add the project to the group.
   c. From the **Method** dropdown list, select an HTTP method.
   d. In the **Amount** field, enter the number of instances of the project to include in the group.
   e. From the **Run In** dropdown list, select one of the following options. These options determine how the projects run among themselves.

      Parallel. Run all instances of the project at the same time.

      Succession. Run instances of the project in succession, one after another.

      Time Lapse. Run instances of the project in the interval you specify in the **Delay** field.
   f. Repeat **Steps a** through **e** to add additional projects to the group.

Managing Groups of Projects

There may be occasions when you need make changes to the projects you have added to a group. The following information will help you manage groups of projects:

- To change the order of a project in a group, use the arrow buttons to move the project up or down in the list.
- To temporarily inactivate a project in a group, check the **Inactive** box. To reactivate the project, clear the box.

- To remove a project from a group, select the file and click the **Delete** button.

Testing Groups of Projects

After you have created a group of projects, you can start testing them.

To test a groups of projects

1. Open Send Master and select **File**, **Automation**.
2. In the **Group Definitions** section, from the **Group** dropdown list, select group to test. The projects in the group display in the **Group Projects** section.
3. Make any needed adjustments to the group, such as change the order of projects in the group, inactive or active projects, and so forth.
4. Click the **Start Projects** button to run the test of projects in the group.

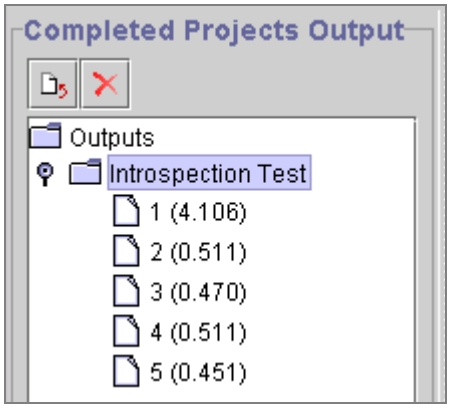
Viewing Test Output

After you run a test on a group of projects, you can view processing information as well as response information for any project in the group.

Viewing Processing Information

After you run a group of projects, the **Completed Projects Output** section displays all of the projects in the group and the instances for each project, in a hierarchical tree format. To expand and collapse a project folder, click the icon to the left of a folder.

When you expand a project folder, the instances for the project display as shown in the following graphic.



Output for the Introspection Test project

Each Page icon represents one of the project instances. The number in parenthesis represents the amount of time elapsed to process the project instance.

To view more detailed processing information about the entire group of projects, select a project, click the **Export the Results to File** button and save the contents as a text file. You can then open the text file and view information, such as the total number of project instances in the group, the total time to process all project instances, processing start and end times, and more. The following example shows the type of output you can view using the Export feature.

```
Count                  : 5
   Total Elapse Time      : 6.049
   Minimum Elapse Time    : 0.451
   Maximum Elapse Time    : 4.106
   Average                : 1.209
   # Processed Per Second : 0.827

     [1] start = 11:50:02.890, end = 11:50:06.996 (4.106)
     [2] start = 11:50:07.006, end = 11:50:07.517 (0.511)
     [3] start = 11:50:07.537, end = 11:50:08.007 (0.47)
     [4] start = 11:50:08.017, end = 11:50:08.528 (0.511)
     [5] start = 11:50:08.538, end = 11:50:08.989 (0.451)
```

Viewing Response Information for a Project Instance

Send Master enables you to view response information for any project instance in a group of projects.

To view response information for a project instance

1. Select a project instance in the **Completed Projects Output** section.
2. Click a project instance. Response information displays in the **Output Information** section.

Sharing Projects and Groups

When you create projects and groups, all data is stored in the smprop.xml file. The location of this file depends on the web server that you are using:

WebLogic:

- C:\bea\weblogic6.1\config\peoplesoft\applications\PSIGW\WEB-INF\classes\com\peoplesoft\pt\sendmaster

WebSphere:

- C:\WebSphere\AppServer\installedApps\peoplesoft\PSIGW\WEB-INF\classes\com\peoplesoft\pt\sendmaster

You can share and reuse projects and groups that you or others have created for other versions of Send Master or that have been used on other workstations, by copying smprop.xml into the Send Master directory. You must rename or delete the existing smprop.xml file before you copy the new file into the directory.

After you copy the smprop.xml file into the Send Master directory, you can access the project and groups in the normal fashion, by accessing them from the **Project** dropdown list in the Project workspace, or from the Group dropdown list in the Automation workspace.

Posting Third-Party Messages to the Integration Gateway using the Simple Post Tool

PeopleSoft provides a simple command line post tool that enables you to use shell scripts to post XML messages from third-party systems to the Integration Gateway. The post tool wraps the incoming messages in the PeopleSoft XML format and post them to the HTTP Listening Connector.

You can also use this tool to batch process the posting of messages to the Integration Gateway.

The topics in this section are:

- Getting Started Using the Simple Post Tool.
- Understanding the Simple Post Tool.
- Posting a Third-Party XML Message to the Integration Gateway.

Getting Started Using the Simple Post Tool

This section describes the location of the Simple Post Tool and the software requirement to use it. It also describes how to launch the tool and the required and optional parameters you must specify. A code example is also included for your reference.

Tool Location

The Simple Post Tool is a Java class with the package name com.peoplesoft.pt.simplepost.SimplePost.

The location of the tool depends on the Web server you are using:

- WebLogic:

c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\WEB-INF\classes\com\peoplesoft\pt\simplepost

- WebSphere:

c:\WebSphere\installedApps\peoplesoft\PSIGW\WEB-INF\classes\com\peoplesoft\pt\simplepost

Software Requirements

To use the tool you must have the Java Runtime Environment (JRE) installed.

Setting Environment Variables

To use the Simple Post Tool, must perform one of the following actions:

- Modify the CLASSPATH so that the start directory is where the simple post tool is located.
- Pass the location of the PeopleSoft classes when you call the Simple Post class. The following example shows passing this information in a WebLogic environment:

```
java -cp "C:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\WEB-
INF\classes" com.peoplesoft.pt.simplepost.SimplePost ...
```

Understanding the Simple Post Tool

This section provides an overview of the Simple Post Tool, including its:

- Usage
- Syntax
- Parameters

Usage

The standard usage of the Simple Post Tool is:

```
com.peoplesoft.pt.simplepost.SimplePost [-options]
```

Syntax

The syntax for sending an XML message from a third-party system to the Integration Gateway is:

```
com.peoplesoft.pt.simplepost.SimplePost -reqnode <requesting node> -msgname
<message name> -url <destination server URL>
-infile <name of input file name to send> -outfile <output file name and path>
-msgtype <message type> -msgver <message version> -destnode <destination node
name(s)> -v <Display debugging
output> -to <timeout value> -?-help <Display help>
```

Note that you enter the syntax as a single line.

The following example shows the syntax with several sample parameters.

```
java com.peoplesoft.pt.simplepost.SimplePost -reqnode KACNODE -msgname
QE_F18_ASYNC -url http://intgateway01/PSIGW/
HttpListeningConnector -infile C:\sendmaster\input\
QE_F18_ASYNC.xml -outfile "C:\Documents and Settings\
JFranco\Desktop\out.xml" -msgtype async -msgver VERSION_1
```

```
-destnode UNDERDOG -v
```

Parameters

The Simple Post Tool parameters you can pass are shown in the following table.

-reqnode	Identifies the requesting node name.
-msgname	Identifies the name of the message you are sending.
-url	Identifies the destination server URL.
-infile	Identifies the name of the XML input file to send. The root node must be name of the message. For example, if the name of the message is SYNC_TEST the root node of the XML input file must be <SYNC_TEST>.
-outfile	Identifies the path and filename where the tool generates the response from the server.
-msgtype	(Optional) Identifies the message type. Valid values: • *Sync*: Specifies that the message you are sending is synchronous. • *Async*: Specifies that the message you are sending is asynchronous. • *Ping*: Tests the application server to make sure it is available and accepting requests.
-msgver	(Optional) An integer value that identifies the version number to apply to the message.
-destnode	(Optional) Identifies the destination node name.
-v	(Optional) Displays any debugging output.
-to	(Optional) Identifies the timeout value. The time value is an integer value that determines the amount of time, in seconds, that the Simple Post class will wait for a response from the server.
-?-help	(Optional) Displays a list of the Simple Post Tool parameters.

Using the Simple Post Tool

This section provides step-by-step instructions for using the Simple Post Tool to post XML messages from third-party systems to the Integration Gateway.

Posting a Third-Party XML Message to the Integration Gateway

To post a third-party XML message to the Integration Gateway

1. Open the Simple Post Tool.

   Windows NT

a. Open a Windows command prompt (DOS).

b. Navigate to the tool location. The location is based on the Web server you are using as described earlier in this section.

UNIX

a. Open a terminal window or shell window.

b. Navigate to the tool location. The location is based on the Web server you are using as described earlier in this section.

2. Enter the following command followed by parameter name and value pairs.

```
java com.peoplesoft.pt.simplepost.SimplePost
```

At a minimum you must enter parameter name and value pairs for:

- -reqnode
- -msgname
- -url
- -infile
- -outfile

3. Press **Enter**.

Glossary

The terms in this glossary are used among multiple Financials and Supply Chain Management applications.

Numbers

401(a)(17) Limits

The limitations on the earnings that may be included in the calculation of benefits under qualified U.S. pension plans.

1st Year Amount

In PeopleSoft Workforce Analytics, 1st Year Amount is an employee-level compensation amount, totaling the calculations for the first calendar year's worth of accounting periods, in a compensation scenario.

A

Abend

Abnormal End (to a process).

ABM (Activity-Based Management)

See PeopleSoft Activity-Based Management.

ABPS (Activity-Based Planning and Simulation)

See Activity-Based Planning and Simulation.

Absence

An absence occurs when an employee is not at work (absent) during a normally scheduled work period. Absences may be scheduled or non-scheduled, compensated or uncompensated, excused or unexcused. An absence may occur for a variety of reasons like illness, family emergency, civic obligations (e.g. Military duty or jury duty), or vacation.

Absence Entitlement

Element which defines the rules for granting paid time off for valid absences, such as sick time, vacation, and maternity leave. An absence entitlement element defines the entitlement amount, frequency, and entitlement period.

Absence Take

Element which defines the conditions that must be met before a payee is entitled to take paid time off.

Accepted Exception

An exception that has been reviewed and validated (see Time Management).

Accommodations

Accommodations are efforts your organization is able to make for employees or applicants with disabilities, such as purchasing special equipment or making structural changes to a work environment.

Account Management

In PeopleSoft Demand Planning, a feature that enables you to divide a centrally held corporate forecast into multiple subsections for easier maintenance and management. These subsections are separate databases that can be distributed to account managers for use and updates, then rejoined with the main database at a later date.

Account

A code for recording and summarizing financial transactions as expenditures, revenues, assets, or liabilities balances. This is a delivered PeopleSoft ChartField, specific use of which is typically defined by the organization during implementation of PeopleSoft General Ledger.

Account Type

A name for one of the different kinds of accounts used in a PeopleSoft General Ledger, such as Asset, Liability, Equity, Revenue, and Expense.

Accounting Class

In PeopleSoft Enterprise Performance Management, an attribute that defines how the particular resource would be treated for generally accepted accounting practices. Inventory denotes whether a Resource will become part of a balance sheet account such as inventory or fixed assets, while Non-inventory denotes that the Resource will be treated as an expense of the period during which it occurs.

Accounting Date

The date that a transaction is recognized as opposed to the date the transaction actually occurred—the **Transaction Date** (although the two dates can be the same). The accounting date determines the period in the general ledger to which the transaction is to be posted. You

can only select an accounting date that falls within an open period in the ledger to which you are posting. The accounting date for an item is normally the invoice date. In PeopleSoft Asset Management, the difference between accounting date and transaction date determines whether prior period depreciation must be calculated, and how much. Accounting Date must be later than or equal to Transaction Date.

Accounting Entry

A set of related debits and credits. An Accounting Entry is made up of multiple ***Accounting Lines.*** In most PeopleSoft applications, accounting entries are always balanced (debits = credits). Accounting entries are created to record accruals, payments, payment cancellations, manual closures, project activities in general ledger, and so forth (depending on the application).

Accounting Entry Template

A user-defined table that controls the use of system-generated accounting lines in the posting processes.

Accounting Split

Method indicating how expenses are allocated or divided among one or more sets of accounting ChartFields.

Accredited Education

Education above the high school level completed in a U.S. college, university, or other educational institution that has been credited by one of the accrediting agencies or associations recognized by the Secretary, U.S. Department of Education.

Accrual

Any hours that employees accumulate for use at another time in the form of earned vacation time or sick leave, for example.

Accrual Basis Accounting

Accounting that records the impact of a business event as it occurs, regardless of whether the transaction affected cash.

Accrual Class Codes

Classes or categories of accruals.

Accrual Type

Defines an accrual such as annual leave or sick leave.

Accumulate Demand

In PeopleSoft Demand Planning, a transfer process function that adds demand quantities for an item to any quantities that already exist for the period.

Accumulator

Element which allows you to combine several elements. For example, an accumulator could consist of all voluntary deductions, or all company deductions, enabling you to accumulate amounts. It allows total flexibility for time periods and values accumulated. See also Time Administration.

Accumulator [Global Payroll]

Element which provides a means for storing the cumulative values of defined items as they are processed. As you make payments, take deductions, and perform calculations, you'll use accumulators to track accumulated amounts, or balances. You can accumulate a single value over time or multiple values over time, as your requirements specify. For example, an accumulator could consist of all voluntary deductions, or all company deductions, enabling you to accumulate amounts. It allows total flexibility for time periods and values accumulated.

Action

In PeopleSoft Deduction Manangement, a task that you perform to obtain information required to resolve a deduction.

Action and Conditions

A process that defines actions and conditions independently of one another and then combines them to create a complete rule (see Rule Creation).

Action Code

In PeopleSoft Engineering, a user-defined code associated with an event/action triggered by the implementation of an engineering change order (ECO). Actions could include analyzing an item's existing quantity on hand, scrapping existing inventory, or modifying current documentation.

In PeopleSoft Product Configurator, a 2-character code that identifies rule types. For example, ***FP*** is the action code for the Finalize Price rule, and ***CN*** is the action code for the Condition rule. The rules control the processing path for configured items.

Action List

An online list of customers who meet predefined credit management criteria. The list also includes appropriate procedures for each action and contact information for the customer.

Action Owner

In PeopleSoft Deduction Management, the individual assigned a task to obtain information to resolve a deduction.

Action Reason

The reason an employee's job or employment information is updated. The action reason is entered in two parts: a ***personnel action,*** such as a promotion, termination, or change from one paygroup to another—and a ***reason*** for that action. Action Reason is used by PeopleSoft Human Resources, PeopleSoft Benefits Administration, PeopleSoft Stock Administration, and the COBRA Administration feature of the Base Benefits business process.

Active Control

A target control requiring that the user validate the budget against the planning targets before submitting it. If the budget totals are not within the tolerance levels, the system indicates that the status is invalid and the user cannot submit their budget until the budget is modified and the amount is within the tolerance range of the planning target.

Activity

In PeopleSoft Receivables and Deduction Management, an action taken on an item, such as creating an item, unposting an item, or writing off an item.

In PeopleSoft Projects, the unit of work that provides a further breakdown of projects—usually into specific tasks. Resources are assigned directly to activities within a project, not directly to projects.

A self-contained task that is part of one or more business processes. Business process maps display the activities that make up the process. An activity consists of steps representing the pages the user needs to complete and events representing the workflow routings triggered by the user's actions.

In PeopleSoft Enterprise Warehouse, the work of an organization and the aggregation of actions used for Activity-Based Costing.

Activity Attributes

Activity Attributes provide pieces of activity information. For example: capacity and performance, cost drivers, cycle time and performance measures.

Activity-Based Costing (ABC)

A methodology that measures the cost and performance of activities, resources and cost objects, assigns resources to activities and activities to cost objects based on their use and recognizes the causal relationships of cost drivers to activities.

Activity-Based Management (ABM)

See PeopleSoft Activity-Based Management (ABM).

Activity-Based Planning and Simulation (ABPS)

ABPS, a feature of PeopleSoft Activity-Based Management, calculates resource demands, new rates, costs, and activity volumes based on demand forecasts. It converts the new

resource demands into new cost requirements at the General Ledger item level to feed as input for budgeting.

Activity Driver

An Activity Driver indicates the amount of demand there is for a particular activity and it is used to assign cost to cost objects. In some instances, an activity driver may represent the yield of an activity.

Activity Fragmentation

The part of the Employee Profile feature that provides information about the number of employees that is involved in completing a particular activity on a full or part-time basis.

Activity ID

A unique 15-character alphanumeric identifier given to each activity within a project. Activity IDs need only be unique within a single project.

Activity List

In PeopleSoft Pension Administration, a checklist used to monitor pension-related activities.

Activity Type

A user-definable identifier for grouping activities.

Activity Type

Also known as Activity Code. A categorization of work effort. Typically work effort is categorized as productive or non-productive; Repair, Maintenance, Enhancement, or Improvement; or Development or Construction. Activity type is usually required to support cost accounting or financial accounting (recording) functions. It may also be required to support some organizational administration requirements such as organizational productivity goals, or employee performance measurement. In some companies, activity type is inferred from job function, work group affiliation, or organization.

Activity Use

An attribute used to describe the behavior of an Activity as defined within PeopleSoft Enterprise Performance Management. A Primary Activity is an activity that is performed for the purpose of directly generating revenue within the course of business. A Secondary Activity is generally performed in direct support of a Primary Activity such as activities related to human resources or MIS.

Actual Base Hours

This defines the number of hours that an employee is expected to work within a given period under analysis within PeopleSoft Enterprise Performance Management. Hours worked in excess of Actual Base Hours are generally considered overtime, while hours worked less than Actual Base Hours would illustrate that the employee is working part-time.

Actual Contribution Percentage (ACP)

The amount of an employee's after-tax or employer matching contributions made in a Section 401(m) plan on behalf of highly compensated plan participants, divided by the employee's annual compensation, or an amount determined in the same manner with respect to non-highly compensated employees. The Base Benefits business process is set up to perform ACP nondiscrimination tests for Section 401(m) plans. See Nondiscrimination Tests and Highly Compensated Employee.

Actual Date

Calendar date in which a punch occurred (see Time Reporting).

Actual Deferral Percentage (ADP)

The amount of salary reduction contributions made by an employee to a Section 401(k) plan for a year, divided by the employee's total compensation for that year. The Base Benefits business process is set up to perform ADP nondiscrimination tests for Section 401(k) plans. See Nondiscrimination Tests and Highly Compensated Employee.

Actual Demand

In PeopleSoft Demand Planning, an **Array** of demand by historical period imported from an external system. The demand figures are determined by imported values and typically include shipments, orders booked, orders booked by requested ship date, or shipments.

Actual Rates

An Actual Rate is the rate that your business currently uses for its business practice.

Actuarial Assumptions

Any assumptions used to calculate an equivalent benefit for an optional form of payment or an alternative retirement date.

Actuarial Valuation

A comparison of a pension plan's assets and liabilities.

Actuarial Valuation Extract

A PeopleSoft Pension Administration data extract containing data that a plan actuary needs in order to determine the plan’s assets and liabilities.

Address Type

A high-level address classification that identifies addresses associated with a **Material Issue**. Examples include Ship To Address, Bill To Address, and Ship Notification Address.

Adjusted

In the Enterprise Planning and Simulation forecasting process, in addition to versions of the statistical forecast, there is an adjusted version of the forecast. Managers create this version by reviewing the forecasts and entering adjustments that cannot be inferred statistically. For example, there may be a promotional campaign next quarter that is expected to boost volume for certain products over several weeks.

Adjusted Demand

In PeopleSoft Demand Planning, an **Array** of demand after adjustments have been made to the actual demand values. The adjusted figures may include both manual and system-generated changes, such as demand filtering and depromotion. The system uses adjusted demand rather than actual demand in the Forecasting Reset process and in the recalculation of model components during period-end processing.

Adjusted Forecast

In PeopleSoft Demand Planning, a **Statistical Forecast** that has been adjusted using management overrides, proration, or summarization.

Adjustment

See **Bill Adjustment** or **Inventory Adjustment**.

Adjustment Voucher

A PeopleSoft Payables voucher that enables you to apply an adjustment to an existing voucher or to relate one voucher to another.

Advice

The Form that employees who choose direct payroll deposit receive in lieu of a check.

Affiliate

A control person of a corporation. Generally, an officer, director, or major shareholder that has the ability to influence the corporate management decisions.

After-tax Deductions

Deductions that reduce net pay. These deductions are subtracted from gross pay after taxes have been taken out. Also called “post-tax” deductions.

Agency

Any Department or independent establishment of the Federal Government, including a government-owned or -controlled corporation, that has the authority to hire employees in the competitive, excepted, and senior executive services.

Aggregated

In Enterprise Planning and Simulation, each period the statistical forecast is calculated automatically by the system. A forecast for each individual product can be computed using history for that product. Then these forecasts can be aggregated (that is, summarized) into forecasts for the product family.

Aggregate Reporting

The ability to report time as a collection or mass. In Time and Labor aggregate time reporting features include the ability to report time in a lump sum, as a pattern, in a range of dates, or for an entire crew.

Aging Data

Updating data from separate sources, and separate dates, to a common date using an annualized factor.

Aging ID

A code representing rules for aging open items.

Alias

Any of several PeopleSoft Pension Administration utilities that look up or calculate employee information.

Allocated

In Enterprise Planning and Simulation, the computed forecast and the summarized forecast are two different versions of the statistical forecast. In addition, the forecast at the product family level can be allocated down to the individual products. Usually this allocation is done in proportion to the calculated product forecasts at that level. This version of the (statistical) forecast is called the allocated or prorated statistical forecast.

Allocated Inventory

The inventory assigned to a specific stock request.

Allocation Manager

Perform allocations using the Allocation Manager. Allocations enable you to distribute revenue, expense, and statistical quantities across business units, departments, and so on. You can allocate budget planning to detail levels so that you may perform detailed budgeting. The type of allocation you select determines the output.

Allocation Manager Rules

In the PeopleSoft Enterprise Warehouse, Allocation Manager rules allow you to specify the basis as well as the target tables for moving, aggregating, or multidimensionalizing your output. Rules use Allocation Manager methods to enrich the PeopleSoft Enterprise Warehouse data. *See* Allocation Manager Methods.

Allocation Manager Methods

There are several methods: Arithmetic Operation, Prorata, and Spread Even. Each method enables you to move and/or enrich output.

Allocations

A process of distributing budget amounts to and from other Budget Centers. Budget amounts are allocated to cover, or offset, the costs in one Budget Center by charging them to another Budget Center. An allocation is also the budget amount that is distributed to or from a Budget Center. A budget amount that is charged to another Budget Center appears as a negative amount. This same budget amount appears as a positive amount in the other Budget Center receiving the allocation. PeopleSoft Budgeting-specific.

Allotment

This is a voluntary deduction from pay. Employees may elect up to two allotments from pay, transmitted to a financial institution to the employee's checking or savings account.

ALM (Asset Liability Management)

See PeopleSoft Asset Liability Management.

Allowances

The amount owed to an employee in addition to base salary and which is not defined as part of gross salary. For example, vacation can be considered an allowance. PeopleSoft Budgeting-specific.

Alternate Account

A feature in PeopleSoft General Ledger that enables you to create a statutroy chart of accounts and enter statutory account transactions at the detail transaction level as required for recording and reporting by some national governments.

Alternate BOM

Identifies the multiple ways in which an item can be produced. The primary production BOM is designated as BOM code 1. By using BOM codes, you can associate up to 98 other alternate BOMs with the item.

Alternate Routing

A routing, usually less preferred than the primary routing, but resulting in an identical item. You can specify up to 98 alternate routings for production routing types by entering additional Routing Codes (greater than 1) for the same routing type.

Alternative Minimum Tax (AMT)

AMT is calculated by adjusting the taxpayer's regular taxable income with a number of tax preference items and adjustments. Tax preference items are positive items increasing

Alternative Minimum Taxable Income (AMTI) and are excluded from regular taxable income. Tax preference items include gain from the exercise of incentive stock options.

Amount Type

In PeopleSoft Workforce Analytics, the Amount Type specifies whether a benefits compensation amount is a value or expense, to the employee or the employer.

Analysis Base

Defined static, historical data used both to seed and compare against proposed budgets.

Analysis Group

A grouping of analysis types. Analysis groups can be used for project analysis and grouping or for mapping analysis types.

Analysis Template

A set of pre-defined reports that you can view and publish online. These templates access data in the Enterprise Warehouse tables, and organize it by function, role and industry. The templates allow you to pivot, sort, rank, drill and chart the data, for your analysis needs.

Analysis Type

A 3-character, user-definable identifier that enables you to label the different types of costs. For example, you might want to track budgeted costs (BUD), committed costs (COM), and actual costs (ACT).

Analytical Applications

See PeopleSoft Analytic Applications.

Analytic Forecasting

Analytic Forecasting is the part of the Planning and Simulation feature that creates forecasts for your business requirements.

Annual Amount

In PeopleSoft Workforce Analytics, Annual Amount is an employee-level compensation amount, totaling the calculations for a full fiscal year's worth of accounting periods, in a compensation scenario.

Annual Declaration Report

The French Annual Declaration report is a payroll report which checks establishment profiles to see whether an establishment has to produce the report, and then calculates the amount of all the social security contributions for this establishment.

Annual Leave

Annual leave is absence from work with pay and must be approved by the employee's supervisor in advance. This type of leave (Plan Type 51) is accrued based on years of service: Full-time Permanent/Full-time Seasonal employees ...0-3 years - 4 hours per biweekly pay period; 3-15 years - 6 hours per biweekly pay period (plus an additional 4 hours in the final pay period of the leave year); and 15+ years - 8 hours per biweekly pay period. Part-time Permanent/Part-time Seasonal employees...0-3 years - 1 hour for every 20 hours worked; 3-15 years - 1 hour for every 13 hours worked; 15+ years - 1 hour for every 10 hours worked. Generally, there is a leave year ceiling of 240 hours on accrual; amounts accrued in excess of the ceiling and not used prior to leave year-end are forfeited.

Annual Shareholders Meeting

A meeting of corporation's directors, officers, and shareholders held for the purpose of communicating the operating and financial results for the prior year, the prospects for the future and major decisions of management.

Annual Workforce Survey by Nationality and Professional Category (Enquête sur l'activité et les conditions d'emploi de la main d'oeuvre)

In France, companies are required to submit the Annual Workforce Survey by Nationality and Professional Category to the Ministry of Labor. This report provides an analysis of the company's foreign workforce, which includes any employee who does not have French citizenship.

Annualized Tax Method

A payroll tax calculation method that divides the tax on an annualized amount by the number of pay periods in the year to find withholding for a given pay period, based on the number of withholding allowances. Annualized is the most common tax method.

Annuitant Amount

The gross monthly annuity a federally retired employee receives.

Annuitant CSA Number

A unique number assigned by OPM for a retired employee.

Annuitant Indicator

A code used to indicate the status of an annuitant appointed to a position in the Federal civilian service. Text for the codes is as follows:

1. Reemployed annuitant - Civil Service/FERS
2. Retired military officer receiving pay
3. Retired military non-officer (enlisted) receiving pay
4. Retired military officer receiving pay and a reemployed annuitant - Civil Service

5. Retired military non-officer (enlisted) receiving pay and a reemployed annuitant - Civil Service

6. Not applicable (none of the above)

Annuitant Indicator (cont)

A. Reemployed Annuitant – FERS

B. Former Annuitant - FERS

C. Retired Officer/Reemployed Annuitant - FERS

D. Retired Officer/Former Annuitant - FERS

E. Retired Enlisted/Reemployed Annuitant - FERS

F. Retired Enlisted/Former Annuitant - FERS

Annuity

A series of periodic payments made to an individual. Under a pension plan, these payments are generally made monthly.

Anti-Dilutive

Typically, options or shares where the price is greater than the current fair market value of the security.

APE (Activité Principale Exercée) Codes

APE codes classify the type of industry or activity your French company is in, such as software, banking or insurance. The APE codes are a normalized set of codes that are required by law and are used in regulatory reporting.

API

An Application Programming Interface (API) is the technology that a software product supplies so you can control it or communicate with it from another application. PeopleSoft APIs enable the user to perform desired actions upon PeopleSoft data without having to know the internal logic or rules of the program.

Applicant Hire Process

The procedure of hiring an applicant who has been tracked and administered in the Recruitment pages. Once you assign an Employee ID, the system uses recruitment data to populate the fields in the Personal Data pages.

Application agent

An application agent is an online agent that is loaded into memory with a PeopleSoft page. It detects when a business rule has been triggered and determines the appropriate action.

Application Designer

The integrated development environment used to develop PeopleSoft applications.

Application Engine

PeopleTools batch processes consisting of a set of defined SQL statements. Application Engine processes is more efficient than COBOL or SQR, since they operate within the database system, and don't rely on external processing.

Application Journal Template

A set of rules and default values to control the creation of journals from accounting entries.

Application Processor

The Application Processor is the PeopleTools runtime engine that controls processing of the application from the time the user requests a panel group from an application menu through the time that the database is updated and processing of the panel group is complete.

Application Server

The application server is the centerpiece of PeopleSoft's three-tier architecture. It utilizes Tuxedo, BEA Systems' transaction monitor, to manage client transactions and provide the business rules and workflow capabilities of PeopleSoft's enterprise applications.

Application Server Domain

The collection of server processes and associated resource managers defined by a single PSTUXCFG configuration file. Each application server domain is configured to connect to a single database. Multiple application server domains can exist on the same server machine.

Appointing Authority

The basis that authorized the appointing officer to effect personnel actions on an employee.

Appointing Officer

Denotes if the employee has appointment authority based on laws and regulations.

Approve Time

The Time and Labor feature that approves all employee daily time before it can be sent to payroll for processing. You can approve time by group or by individual employee. You can also unapprove previously approved time.

Approving Official

Individual with the delegated authority responsible for signing the action(s) taken on an employee.

Array

An ordered grouping of data by period and year. PeopleSoft Demand Planning uses arrays in forecasting demand.

Array

Element which enables you to extract information based on a column value. One way of thinking of an array is that it is a SQL statement that retrieves data from an existing table.

Array Dimension

Determines which inventory-stocking possibilities are included in a **Cube View**. This standard one-level dimension consists of the key fields that include, for example, order quantity, safety stock, and turn rate.

Arrears Balance

An amount owed to either the employer or employee, usually the result of a deduction not fully taken.

Ask Price

The price at which someone who owns a security offers to sell it; also known as the asked price.

As-of-Dated

Refers to a snapshot of the data at a given point in time.

Asset Assignment

A streamlined means of associating project costs to assets or asset profiles within PeopleSoft Projects.

Asset Budgeting

Budget for planned asset acquisitions and the associated depreciation expense that can be associated with a Capital Acquisition Plan (CAP).

Asset Catalog

A list of asset profiles which includes information about that asset type, including Cost, Life, Salvage Value, Depreciation Method, Currency Code, and Asset and Depreciation Account.

Asset Category

A standard group of assets. Typical asset categories include Furniture and Fixtures, Machinery and Equipment, Land, Buildings, Leasehold Improvements, and the like. These generally correspond to General Ledger asset accounts. Assets in one category usually share some depreciation characteristics, such as estimated service life and depreciation limits.

Asset Class

An asset group used for reporting purposes. It can be used in conjunction with Category to refine asset classification.

Asset Liability Management

See PeopleSoft Asset Liability Management.

Asset Life

The number of years an asset will depreciate, after which time it might be kept or sold for its Salvage Value. Also *see* Useful Life.

Asset Profile

A template that contains standard depreciation criteria for an asset type and its corresponding asset books. You can use the information in asset profiles as default values when adding assets.

Assignment of Life Insurance

Effective 10/3/94, Federal employees can assign their Basic, Option A and Option B insurance to another person(s), firm(s), or trust(s); Option C is excluded. The assignment of benefits transfers ownership of the FEGLI coverage to the assignee(s). The insured no longer has control over his/her insurance coverage and can no longer designate beneficiaries. Assignment is irrevocable. Either all or none of the insurance can be assigned. Assignment does not have to be to the same person or firm. Assignments must be made in percentages of total insurance versus an assignment of Basic Insurance to one person and Option A to another. Additionally, terminally ill employees can assign their insurance to a Viatical Settlement Firm in exchange for cash (approx. 60% - 85% of the face value of the coverage). Life Expectancy is usually 24 months or less for a Viatical Settlement Agreement.

Assignment Type

This defines the behavior of the object, (resource, activity, or cost object) within PeopleSoft Activity-Based Management. If the object is identified as a source then costs may be allocated from that object to another object, which must be identified as a target. If an object ID is identified as a target it may be allocated costs from another object ID but may not allocate costs. An object ID can be both a source and a target, thereby having the functionality of each.

Associated Primary BOM

With multiple outputs, it's possible that a given co-product can be created in more than one way – in other words, an item is a co-product on more than one items' primary BOM. By assigning an associated primary BOM to a co-product, you are telling the system which BOM to use in exploding the co-product to the next level.

AT Section

In France, this stands for Section Accident du Travail, or Work Accident Section. It is information needed to identify the establishment risk code for insurance purposes.

ATP Reserved Order

An order that has been promised against future supply. The user has an obligation to the customer to fulfill the order quantity by a certain date. ATP-reserved orders are also referred to as *promised orders.*

Attendance

A component of time reporting application whose purpose is to apply business rules related to Benefit Entitlement and Administration and Organizational Administration to time reported as worked or not worked, and to satisfy a variety of reporting needs.

Attendance Reporting

A Time and Labor report that indicates an employee's attendance record. It includes sick leave, vacation time, and other leaves taken.

Attribute

An attribute is an element within a dimension. For example, the element "Store" is an attribute of the dimension "Geography" for the retail industry. An attribute is also a column heading on an analysis and reporting template.

Audit Trail

See Drill-Back Calculation.

Auditor

Person designated to review expense sheets and cash advances before payment.

Automatic Revision Incrementing (Auto Rev)

The ability to automatically set up revision control and generate revisions for revision-controlled items at the business unit level. This includes setting up a revision scheme or a predetermined, ordered list of revision names.

Automatic Spouse Benefit

A joint and survivor pension benefit provided without any actuarial reduction to a pension benefit. The automatic benefit is a n% joint and survivor; the employee is still entitled to choose any optional form of payment and any beneficiary for the remainder of the benefit.

Availability Date

The date a lot becomes acceptable for fulfillment in PeopleSoft Inventory or for consumption in PeopleSoft Production Management. (Availability Date = Creation Date + Availability Lead Time)

Available to Promise (ATP)

The projected supply of a product less the actual demand, which informs the sales and marketing department of the products that can still be sold without modifying the master schedule. ATP isn't cumulative – it's calculated for each period.

Average Daily Balancing

A feature in PeopleSoft General Ledger that enables you to target the ChartFields on which you base average balance calculations, summarize amounts for selected ChartField values according to your reporting requirements, and define the periods for these calculations.

Used by the financial analytic applications in Enterprise Performance Management. For a reporting period (usually monthly) this refers to the average daily balance of an account as opposed to the month-end-balance, which is the balance as of the last day of the month.

Average Daily Balance Ledger (ADB_Ledger)

In the PeopleSoft Enterprise Warehouse, the Average Daily Balance Ledger table (PF_ADB_LEDGER_F00) is similar to the functionality of the PF Ledger table (PF_LEDGER_F00), in that it too supports reporting. However, the Average Daily Balance Ledger is used for average daily balances. It is a table that is used mostly for processes associated with the financial services industry.

Average Inventory

In PeopleSoft Inventory Planning, one half of the average lot size plus the safety stock when demand and lot sizes are expected to be relatively uniform over time. When demand and lot sizes are not uniform, the stock level versus time can be charted to determine the average.

Average Price

The average price derived from either the bid and ask prices (for bid/ask/average) or from the high and low prices (for high/low/average).

Average Static Calc Flag

In PeopleSoft Inventory Planning, a method used with static policies. The average method sets the static policy equal to the weighted-average, time-phased policy over the next argument periods.

Award

A special payment to an employee for certain prescribed kinds of activities or accomplishments.

B

Back Pay Interest

Under certain circumstances, an employee can be eligible to receive additional pay relative to a delayed receipt in salary caused by administrative error in processing a personnel action. The U.S. Office of Personnel Management has established guidelines for Federal agencies on when and how to make these calculations.

Background Process

Any task or process that is grouped with another and runs in the background. Background processes are usually scheduled to run on a regular basis. All background processes are executed through process-specific COBOL programs run outside the Windows environment.

Backlog Reason Code

An identifier indicating the reason an item could not be shipped. Example codes might include out of stock, discontinued, or seasonal.

BAD Forecast Ratio

In PeopleSoft Demand Planning,the maximum acceptable value of the ratio of the and the base component (Standard Deviation/Base Component). When this value is exceeded, the system automatically resets forecast model parameters. The higher the value, the less likely it is that the system will reset the parameters. In most organizations, a BAD ratio of 1.00 or lower is appropriate for most items.

Balance Segmentation

Balance Segmentation is used in Funds Transfer Pricing to divide balances in deposit accounts between core (stable) and non-core (volatile) segments. Core funds represent the minimum balances that are retained on a long-term basis, building a relatively reliable source of funding to the bank. Non-core funds are temporary in nature due to their volatility caused by customer preferences for liquidity, and cannot be utilized on a long-term basis.

Balance Type

Balance Type is a lookup code used to define the type of instrument balances that will be stored in the PeopleSoft Enterprise Warehouse and processed by the analytic applications. Examples of different Balance are Current Balance, Average Daily Balance, Period Ending Balance, or Commitment Balance.

Balanced Scorecard

See PeopleSoft Balanced Scorecard.

BAM

Business Analysis Model. XXX I think this term is incorrect because we use BAM to refer to the application. If we were referring to the business analysis model, we would say BAM model (that is, Business Analysis Modeler model.)

BAM Model

The BAM database published from the template. The model contains both the data and analytic structure used in the application. The BAM database is physically separate from the Enterprise Warehouse database. Data is sent to the model through migration processes.

BAM Template

A file created using BAM design tools, representing the model prior to its creation as a database. This file has an extension of .MDL. This file is published to a BAM database once the model design process is complete. Each application using BAM will deliver templates which the customer will review and publish to a database in their environment.

Bank Identification Number (BIN)

In PeopleSoft Payables, a part of the bank information that identifies business unit banks.

Base Budget

The initial budget defined by the Budget Coordinator. The base budget is distributed as a starting point for Budget to review and edit. The base budget can be zero-based or incremental.

Base Compensation

In PeopleSoft Workforce Analytics, Cash Compensation that is typically categorized as fixed. It includes base pay and shift differentials as well as associated merit, equity, and step increases.

Base Currency

Base Currency is used to consolidate and report financial results of a multinational company. When a company transacts its business operations in different transaction currencies, those currencies are translated to the base currency for reporting purposes.

Base Currency Equivalent (BCE) Amount

If the monetary amount is in a currency other than the base currency, either the Extract-Transform-Load (ETL) process or the Multi Currency Engine can be used to convert the monetary amount to the Base Currency Equivalent (BCE) Amount.

Base Factor

In PeopleSoft Demand Planning, an element of a smoothing constant simulation set that controls base component smoothing in the Model Reset Simulation process.

Base Metric

Metric found on a fact table. A base metric usually contains an aggregate operator, for example “sum” or “count”.

Base Pay

A pay component included in the job comp (job compensation rate) calculation. It is pay for a regularly assigned workweek. For example, you can set up a regular hourly rate plus a shift rate, a union-negotiated rate for hazardous work, and so on.

Base Pay Structure

A PeopleSoft Workforce Rewards module you use to create or revise pay structures, and to assess the cost and impact of implementing new structures.

Base Time Zone

Customer defined time zone used for converting reported time to a common time zone for ease of applying rules (see Time Administration).

Batch

Batch systems are used when realtime updates are not needed. Batch-oriented data collection applications, developed in-house or by a third-party vendor, produce transactions that are collected in an ASCII text file. The text file is fed to a PeopleSoft SQR program that loads the transactions into the database.

Batch Processes

Any of the background programs in the client/server environment of PeopleSoft applications. Batch processes perform operations—such as pay confirmation, deduction calculation, and so forth—on groups of records, and are usually scheduled to run on a regular basis. You run these processes from the Process Scheduler, and they are executed through process-specific COBOL programs.

Before-Tax Deduction

Deduction that reduces net pay and FWT taxable gross, applied prior to the calculation of federal and state/provincial withholding taxes. Also called “pre-tax” deductions.

Begin Calc Date

The date on which PeopleSoft Asset Management begins to deduct from an asset's life.

Begin Depr Date

The date on which PeopleSoft Asset Management begins to calculate depreciation for an asset. Begin Depr Date is calculated using In-Service Date and Prorate convention.

Benchmark Job

In PeopleSoft Workforce Analytics, this refers to a Job Code for which there is corresponding salary survey data from published, third party sources. Jobs for which there is no corresponding salary survey data are referred to as non-benchmark jobs.

Benefit Commencement Date (BCD)

The date on which a pension payee elects to begin receiving payments.

Benefit Deduction

Any amount taken from an employee's pay check to offset all or part of the cost of the employee's benefits.

Benefit Eligibility

The PeopleSoft Pension Administration function that determines if an employee is eligible for retirement or ancillary benefits. A plan may have several retirement types—normal, early, late, death, and disability—each with its own eligibility criteria.

Benefit Entitlement

Any rules governing the circumstances under which employees are entitled to receive certain benefits. Typically, entitlement to benefits is based on type of employee (for example, full time, part time, occasional), length of employment, and specific rules which apply thereto, i. e., work group affiliation, and compensation base. Other criteria may also apply, such as reasons-for-claiming or job performance.

Benefit Formula

The formula that determines a participant's pension benefit in a defined benefit plan, as well as the PeopleSoft Pension Administration function that calculates the benefit.

Benefit Group

Part of a group of defaults assigned to job codes. Benefit group may include medical, dental, and health benefits dependent on individual company parameters.

Benefit Plan

A specific benefit within a plan type. For example, your company's life plan type might include benefit plans of one times salary, two times salary, and three times salary.

Benefit Plan Type

Any category of benefit, such as health, life, or savings.

Benefit Program

A set of benefits and deductions valid for an employee or group of employees. A single company may have any number of programs. An individual employee may belong to only

one program; the deductions and benefits contained in that program are the only valid deductions and benefits for that employee.

Benefit Tables

Any of the tables that contain employee benefits information. These are often relevant to payroll processing.

Benefits Base

The salary used for benefit calculations. The benefits base will be either the employee Annual Rate or Annual Benefits Base Rate.

Benefits Compensation

In PeopleSoft Workforce Analytics, Benefits Compensation is value associated with employment benefits. It can include benefits types for Health and Welfare (Medical, Life Insurance), Retirement (annuities, savings plans, pensions), and Paid Time Off (Vacation Leave, Sick Leave). Benefits compensation is sometimes fixed, and sometimes variable, depending upon the benefit type.

Betriebszählung (Company Statistics Report)

Also called the OFIAMT report. This report provides statistics required by the Swiss Federal Department of Statistics (BFS).

Bias Signal Limit

In PeopleSoft Demand Planning, a number between one and six that indicates how many **Forecast Period** to test for bias. If the bias test is violated, the system records a **Tracking Signals** error in the period up to the number of periods determined by the bias signal limit.

Bias Test

In PeopleSoft Demand Planning, a forecasting test that sets the limit for tripping a **Tracking Signals**. The lower the value, the more likely it is that a tracking signal is set.

Bid Price

The price a prospective buyer is prepared to pay at a particular time for trading a unit of a given security.

BIF file

This is the bulk insert file (input.bif) used with the Verity search engine to specify the documents to be submitted to a collection (search index). It contains a unique key, document size (in bytes), field names and values, and document location in the file system.

Bilan Social Report

See Employee Survey Report.

Bill

In PeopleSoft Billing, any group of bill lines.

Bill Adjustment

The process of making credit or credit and rebill adjustments to an invoiced billing activity.

Bill By Identifier

The Bill By Identifier is used to define how billing activity is grouped when added to a bill through the billing interface or the Populate Billing process.

Bill Header

The record containing information that pertains to the bill as a whole. Each bill has a unique bill header that identifies it within the system.

Bill Inquiry Phone

Bill Inquiry Phone is the number printed on your invoices for your customers to call if they have any questions about their bill.

Bill Line

The basic unit of billing activity representing a billable charge, including the charge identifier, quantity, price, and any other information regarding an individual transaction. Every bill line is related to a bill header that may have one or more bill lines related to it.

Bill Search

A method of finding a bill or bill line when you don't have enough information to call up the bill directly. ***Customer Bill Search*** enables you to locate a bill by Customer Name. You can also choose other parameters to limit your search. With ***Bill Line Search*** you first search for a particular bill and then a line on that bill. Parameters for bill line search include Reference, Date, and Amount.

Bill Source

The point where billing activity originates. Bill sources may be external to the system (imported through the billing interface) or entered directly online. Examples of bill sources include order management, project costing, and contract administration.

Bill To Customer

A customer who receives an invoice.

Bill Type

A category of billing activity variety. Examples of Bill Types include standard and custom order activities.

Bill Update

The process that adjusts bills that have either been entered manually or generated within the system.

Billable Indicator

A status flag that identifies an item as eligible for billing to a customer.

Billback Discount (BB)

A per unit discount which typically requires a customer to perform one or more merchandising activities to receive the discount. A BB discount is not deducted from the customer invoice, but once the customer performs the merchandising activity, a sales representative or broker can approve payment for the discount amount. Billback discounts can originate from a National Allowance or Customer Promotion, and are passed to PeopleSoft Order Management for informational purposes only. Billback discounts are recognized as a liability when the product is shipped.

Billing Location

A number identifying a customer address. Each customer may have multiple locations, but must have one *Primary Location* at which you contact them.

Blackout Period

The period of time, determined by the company, which prohibits certain activity in the company stock. Blackout Periods can affect the trading of some key individuals or can be placed on the entire company.

Bonus Tax Method

Annualizes your year-to-date earnings by multiplying them by the number of pay periods in the year. This method is used for Canadian tax processing.

Book

In PeopleSoft Asset Management, a data location storing financial information—like cost, depreciation attributes, and retirement information—on assets.

Borrow/Loan

The temporary reassignment of an employee to other task reporting or compensation requirements to allow the business to meet unexpected, short-term, fluctuations in staffing or work load. Typically, this kind of reassignment is done informally at a local level, where HR isn't involved and a new job record isn't created. Companies may have specific rules about how long an employee may be borrowed/loaned, how and where productive, non-productive, and compensated absence time will be charged, and what business rules to apply to the borrowed employee's time for the purpose of compensation and benefit entitlement and administration. See also Casual work Assignment.

Bracket

Brackets are a way to look up and retrieve database table values. After you've defined a table, the system finds a corresponding row on that table and returns the value of the bracket. The result is then available for use in other items such as formulas.

Branch

A tree node that rolls up to nodes above it in the hierarchy, as defined in the Tree Manager.

Branch Of Military Service

Identifies, if any, military service in which the employee served.

Breadcrumbs

Breadcrumbs show the navigation path to the current web page location. As you drill down through the different levels of the registry, a "breadcumb trail" appears that shows the path you've selected. Each registry level is separated by an angled brace (>), and you can select any level to navigate directly back to that level.

A typical Breadcrumb would look like this:

```
Home > HR > Administer Workforce > Benefits
```

Break Funding

Charges assessed for mortgages that are paid off before maturity. In the Funds Transfer Pricing (FTP) application, Break Funding charges are factored into the transfer price for a loan that may be prepaid.

Break in Service

A period of time for which an employee does not meet stated service requirements.

Break Price

The price used to determine which options are eligible for repricing. For example, if the break price is $36, then all outstanding option with a grant price of $36 and greater are eligible for repricing.

Break Punch

An in/out punch of when a time reporter takes a break.

Brokers

Individuals or organizations who buy and sell securities. Often they are account executives who work for firms registered with the Stock Exchanges and the SEC. Unlike Transfer Agents, (who are not responsible for sales) Brokers do not maintain records on all your company's certificates. They maintain only sales records and stocks for their clients.

BSC (Balanced Scorecard)

See PeopleSoft Balanced Scorecard.

Budget Activity

A type of activity peformed using PeopleSoft Budget Planning. Budget activities include Line Item Budgeting, Line Item Mass Adjustments, Budget Allocations, and Position Budgeting. PeopleSoft Budget Planing-specific.

Budget Amount Ledger

Stores budget amounts and is updated by posting budget entries, transfers, and adjustments.

Budget Analyst

A role within PeopleSoft Budgeting. Budget Analysts are typically people within an organization responsible for reviewing and analyzing a prepared budget before submitting it to the Budget Coordinator. PeopleSoft Budgeting-specific.

Budgetary Account Only

An account used by the system only and not by users; this type of account will not accept transactions. You can only budget with this account. Formerly called "system-maintained account."

Budget Category

A set of related expenses that are accumulated for proposal budgets and reporting to a sponsor. The estimated cost for a set or class of accounts.

Budget Category

Numeric/alpha identification given to each category of positions.

Budget Center

In PeopleSoft Budgets, any entity responsible for producing or reviewing budget data. For example, a Budget Center might be the individual departments responsible for producing budgets.

Budget Center Dimension

In PeopleSoft Budgets, the dimension by which you distribute budget data. If you budget by department, your department dimension will be your Budget Center Dimension. You'll assign Budgets Users to the nodes and detail values on the tree you use to build your Budgets Center Dimension.

Budget Check

In commitment control, the processing of source transactions against control budget ledgers, to see if they pass, fail, or pass with a warning.

Budget Check Override

Selective suspension of Budget Processing. With this feature you can override the controlled budget for a transaction that failed budget checking due to insufficient funds; or override the tolerance limits for a transaction rejected due to exceeded tolerance limits. When you push the Override button, the system flags the transaction to allow the Budget Processor to process successfully regardless of available funding. You can cancel the override any time before the Budget Processor is run by clicking the Cancel Override button.

Budget Control

In commitment control, it ensures that commitments and expenditures don't exceed budgets**.** It enables you to track transactions against corresponding budgets and abort a document's cycle if the defined budget conditions are not met. For example, you can prevent a purchase order from being dispatched to a vendor if there are insufficient funds in the related budget to support it.

Budget Coordinator

A role within PeopleSoft Budgeting. Budget coordinators are responsible for monitoring the budget process. The Budget Coordinator is typically located within an organization's central budget office and builds the budgeting model. PeopleSoft Budgeting-specific.

Budget Detail

A level of itemization that when combined makes up a major budget category.

Budgeted Rates

In PeopleSoft Activity-Based Management, the rate your organization uses based on the budget.

Budget Error Exception

A transaction that fails budget checking, causing an Error or Warning to be issued. See **Error Exception** and **Warning Exception**.

Budgeting Functions

PeopleSoft Budgeting's six main action categories, including: system administration, budgeting setup, budgeting preparation, budgeting analysis, data integration and my profile. Your user role determines how many of these functions display and are available.

Budgeting Model

The framework for an organization's budget development process. Business unit defines a Budgeting Model. The Budget Coordinator typically defines the model and includes the time period of a budget cycle, time period for phases within a budget cycle, the sources of data that will be available to budget users, the methods that will apply to line-item budgets, and other budget options and control parameters. PeopleSoft Budgeting-specific.

Budgeting Type

Associated with the budget ledger type set definition, a budget type is an indication of whether the organization uses a standard budget ledger, project budget ledger, or controlled budget ledger for budgeting.

Budget Justification

Written explanation further defining the what and why of a budget category.

Budget Period

The period in which you define plans to meet your organizations training requirements.

The interval of time (such as 12 months or 4 quarters) into which a period is divided for budgetary, and reporting purposes. The ChartField allows maximum flexibility to define operational accounting time periods without restriction to only one calendar.

Budget Phase

In PeopleSoft Budgets, a span of time during which a budget or portion of a budget is to be completed. You'll filter dimensions, assign alternate Budgets Users, enable Position and Asset budgeting, and specify Budgets User notification options at the Phase level.

Budget Plan

In PeopleSoft Workforce Rewards, when working with a Compensation Planning BAM model. A budget plan is a rollup of like compensation rules. For example, for base pay rules budget plans are a rollup of values for like Action Reasons. For variable pay rules budget plans are a rollup of the values for like Variable Compensation Plan IDs.

Budget Preparer

A role within PeopleSoft Budgeting. Budget preparers are typically people within an organization responsible for developing the detailed budget for a Budget Center and submitting it to a Budget Reviewer or Analyst for review and approval. PeopleSoft Budgeting-specific.

Budget Reviewer

A role within PeopleSoft Budgeting. Budget reviewers are typically people within an organization responsible for reviewing and approving a prepared budget submitted by a Budget Preparer. PeopleSoft Budgeting-specific.

Budget Seeding

Represents a new budget or forecast, such as historical data that is manipulated to develop a more current representation for a proposed budget. Uses detail data as the budget seed or basis to create the base budget that represent the level of detail in which budget numbers are prepared.

Budget Translation Trees

Trees translate (summarize) source transactions into the appropriate levels for processing against control budgets. This is because you usually budget above the level of your source transaction ChartFields on a tree.

Budget Type

Indicates whether a budget is for expenditures or revenues.

Budget Warning

See **Warning Exception**.

Budgets User

In PeopleSoft Budgets, any user who needs to gain access to the Budgets. You'll designate Budgets Users on the Budgets Users page through the Coordinate Budgets window. You'll also assign these users to the tree representing your Budget Center Dimension.

Budget View

A user-defined view where selected dimensions, columns and rows of data determine the layout of line-item budgets affecting the view or entry of data.

Budget Year

The institutionally defined, consecutive, 12-month period to which a financial transaction or summary applies.

Build Option

A detailed PeopleSoft Planning model that specifies a method of building an assembly item. This model specifies the routing, resources, and materials that are necessary to produce the item.

Built-in function

Prior to PeopleTools 8.0, there were only built-in functions, like FetchValue, ScrollSelect, etc. A built-in function, in your code, is on a line by itself, and doesn't (generally) have any dependencies. You don't have to instantiate anything before you can use a built-in.

Business Interlink Definition

A definition encapsulating an external Transaction or Query and providing a set of generically typed input/outputs that can be assigned to PeopleCode variable or Record Fields at runtime. A Business Interlink Definition is added to the Application Designer's objects at the same level as Fields, Records, Panels, etc.

Business Interlink Design-Time Plug-in

An XML file that, when coded for an external system, encapsulate that external system and provide a catalog of Transactions, Classes and Criteria specific and meaningful to that external system.

Business Interlink Framework

The framework for integrating any external system with PeopleTools application objects. It is composed of the following components:

1) An External System, 2) Generic definitions for a Transaction/Query command interfaces, 4) Business Interlink Definitions, 4) Business Interlink Plug-in.

Business Interlink Object

An instantiation based on a Business Interlink Definition. Actual data can be added to the inputs of the Business Interlink Objects once the appropriate bindings are provided. The Business Interlink Object can be executed to perform the external service. Once a Business Interlink Object is executed, the user of that object can retrieve the outputs of the external service. The Business Interlink Objects use buffers to receive input and send output. When a Business Interlink Object is executed, the transaction/query/class associated to the Business Interlink Object will be executed once per each row of the input buffers corresponding to the input Records. If there is only one row, after appropriate substitution by the driver, it is executed only once.

Business Interlink Runtime Plug-in

A set of C++, Visual Basic, or other high-level language methods that, when coded for an external system, encapsulate that external system and provide the execution methods to match the Business Interlink Design-Time Plug-in. (The catalog of Transactions, Classes and Criteria provided by the Design-Time Plug-in can also be provided by the Runtime Plug-in.)

Business Objects

A way of identifying those mass changes that have been designed to be referenced by a flexible formula and provide them with a shorter name to simplify the creation of flexible formulas.

Business Planning

The type of planning that focuses on elimination activities that are not needed by changing the drivers.

Business Rules

Rules that can process information differently depending on the values of data in the PeopleSoft Enterprise Warehouse.

Business Unit

A corporation or a subset of a corporation that is independent with regard to one or more operational or accounting functions. PeopleSoft General Ledger business units typically comprise individual entities for accounting purposes.

Business units in PeopleSoft Projects represent operational structures but not necessarily independent financial units.

PeopleSoft Payables business units are either *Vouching* (have payables accrued to them) or *Charge to* (have voucher expense distributions charged to them), and pass journals to general ledger units.

PeopleSoft Purchasing business units share vendor, purchase order, and receiving information with PeopleSoft Payables units in the same SetID.

A PeopleSoft Inventory business unit is a storage facility that maintains its own replenishment and costing methods, as well as its own definitions and guidelines.

The Manufacturing business unit must be identical to the Inventory business unit in order to link the manufacturing and inventory processes.

The Order Management business unit controls certain order processing parameters (tax and freight calculation methods, base currency, credit card hold options, and so on) for its associated PeopleSoft eStore and Mobile Order Management merchant variants.

Business Unit Audit List

One or more business units specifically targeted for expense report and cash advance audits.

Buying Agreement

You can structure flexible and easy-to-use buying agreements for customers or groups of customers. You can set up maximum amounts and specify the minimum dollar value per order placed against it. You can automatically generate sales orders or create sales orders online from buying agreements. Rebate and penalty calculations can be implemented for buying agreements.

C

Cafeteria-Style Benefits

Any programs offering several benefit plans from which participants make elections. Cafeteria-style benefits may or may not include flexible credits.

Calculation

In PeopleSoft Pension Administration, the determination of a participant's pension benefit.

Calculation Rule

Criteria for calculating benefits, including as-of dates for age, service, premium, and coverage calculations; rounding rules; and minimum and maximum coverage amounts. Any number of program and plan combinations can use a single set of calculation rules.

Calculation Rule [Global Payroll]

Any rule you develop using combinations of elements to command the system to perform a type of calculation.

Calendar

In PeopleSoft Manufacturing, a list defining the days your enterprise is available and the hours of operation for each day. The system first looks to see whether you are using a work center specific calendar. If none is defined, it looks at the production calendar. If no production calendar is defined, planning and scheduling functions base start and due dates on a five-day workweek.

In PeopleSoft Demand Planning and Inventory Planning, a list defining the start and end dates for each time-phased period. It also contains daily weights for distributing raw data into different period buckets.

In PeopleSoft General Ledger, your accounting calendar defines the time periods to which you post transactions for different ledger group and business unit combinations. You can have multiple calendars, so you can keep a calendar for actuals, another for budget and forecast activity, and still others for special reporting or transitional needs.

Calendar Group ID

Allows you to group together multiple Calendars that you want to run together at the same time. It also controls the order in which the Calendars are processed. You can only group calendars together that are for the same country (based on pay entity country).

Calendar Scope

A time period type (Day-Factored, Month-Factored, or Week-Factored) for use in building your time period calendar.

Canada Academic Teaching Surveys

Statistics Canada requires that all Canadian universities (all degree granting institutions) produce full-time and part-time *Canada Academic Teaching Surveys*. These reports are a legislative requirement. PeopleSoft HRMS 8 provides you with the functionality to code HRMS information using Statistics Canada codes and create both the full-time and part-time Academic Teaching Surveys.

Canadian Industrial Sector

The Canadian industrial classification code with which employees are associated for Canadian employment equity reporting purposes.

Canadian National Occupational Classification (NOC) Codes

NOC codes are occupational classification codes for Canadian companies provided by the government.

Canadian Standard Occupational Classification (SOC) Codes

SOC codes are occupational classification codes for Canadian companies provided by the government.

Cancellation

A process that terminates stock fulfillment requests, allowing reserved and allocated items to be returned to inventory.

Cancellation

In the context of an employee stock plan, a transaction (usually triggered by a specific event, such as a termination of employment) in which outstanding securities are declared void and inactive and returned to the pool of securities reserved for issuance under the plan or retired.

Candidate Keys

In PeopleSoft Demand Planning, elements of data that can be used to construct the **Forecast Item** key field at different levels of the forecast.

Capacity Rate

A rate you assign to a capacity cost object. This enables you to track and report on excess capacity.

Capacity Fence

A time fence that indicates that date and time after which PeopleSoft Enterprise Planning or Production Planning solvers ignore capacity violations. The solvers do not use this date in processing capacity violations.

Capacity Multipliers

A multiple used in PeopleSoft Enterprise Planning and Production Planning to determine the available capacity on a resource. Since a capacity multiplier is effective-dated, you can use it to vary the resource's available capacity over time.

Capital Acquisition Plan (CAP)

A method of projecting and tracking capital expenditures for a project. Budgeted assets and actual expenditures can be associated with a CAP Plan so the owner can track planned against actual costs.

Capital Gain

The difference between an asset's purchase price and selling price, when the difference is positive. Capital gains can be either short-term (where the capital asset was held for 12 months or less) or long-term (where the capital asset was held for 12 months or more).

Capital Gains Tax

A tax on profits from appreciation in owned real property, recognized at the time the property is sold; real property includes owned company shares.

Capitalization

The total types and amount of the outstanding securities that have been issued by a corporation. Generally includes both equity and debt securities.

Capital Markets Instrument

In the financial services industry, Capital Market Instruments are assorted financial instruments issued by organizations to raise capital for funding operations. Participants are made up of interested parties that choose to supply or acquire the capital funding through such vehicles. Derivatives, debt instruments, equities and foreign exchange instruments that are traded in highly liquid markets represent the instruments. In the PeopleSoft financial analytic applications, Capital Market securities refer to instruments that are bought/sold by the institution for its own investment account. The capital markets set the product prices and interest rates.

CAP Sequence Number

The number that distinguishes a small project belonging to a CAP plan. Budgeted assets can be associated with an overall CAP Plan and a CAP Sequence, if that level of detailed tracking is desired.

Carry-Forward

Residual contributions that remain in a stock purchase participant's account after the purchase of shares that are used toward future purchases.

Carrying Cost

In PeopleSoft Inventory Planning, a value that shows the cost associated with holding a dollar of inventory for one year. The value is presented as a percentage.

Case Officer

In Germany employees in your company are designated as Case Officers, and have responsibilities for handling health and safety incidents.

Cash Balance Accounts

The PeopleSoft Pension Administration function that tracks the activity in an employee's hypothetical account under a cash balance plan.

Cash Balance Plan

A defined benefit plan designed to look like a defined contributory plan. The plan periodically credits a percentage of pay to each employee's hypothetical account.

Cash Compensation

In PeopleSoft Workforce Analytics, Cash Compensation is a component of direct compensation. Cash Compensation consists of direct cash payments made to an employee for base compensation and short-term variable compensation.

Cash Exercise

At the time of exercise, the optionee is required to pay in cash the total option price plus any withholding taxes due to the company.

Cash Flow Generator

This is a support module for the PeopleSoft financial services analytic applications. It generates actual and projected cash flows for financial instruments by using output from the other support modules, such as loan prepayment rates, deposit runoff rates, product pricing indices, discount rates, and product definitions (such as start and end dates, balance amount, interest rate, term, payment dates, repricing and compounding frequency, and accrual basis) to generate the cash flows. The Financial Performance Measures module accesses the cash flow results to calculate the required financial measures.

Casual Preparer

An additional user role at the lowest level of budget preparation for a budget center. This user performs the same activities as the Budget Preparer role when access is granted. The system does not, however, enable the Casual Preparer role to define their own private views for line-item budgeting.

Casual Work Assignment

The temporary assignment of an employee to a work position or location to meet the needs of the business. Typically, there is no Human Resource activity to support the work assignment (that is, a new Job record is NOT created). Often compensation rules that accrue to the temporary assignment override the compensation rules that apply to the employee's normal work assignment. See also Borrow/Loan.

Catalog

The list of transactions, classes, and queries used to interface to the external system. Integration users are presented with this list when they pick the type of Business Interlink Plug-in they are going to use. There are four types of catalogs: transaction, class, operator, and configuration parameter.

Catalog

A way of organizing your training courses into classifications for increased flexibility. Catalogs consist of categories and subcategories.

Category

Categories are the primary level of a two-tier structure of training courses. Categories can consist of subcategories that provide further course definition.

Category Tree

A hierarchical structure that groups products by category to control how they are displayed in PeopleSoft eStore web pages. Used also by Mobile Order Management to enable product information to be accessed by a wireless device.

CBM

See PeopleSoft Customer Behavior Modeling.

Census Metropolitan Area (CMA) Code

In Canada this code is prescribed by the government and refers to the area of an urbanized core with a population of at least 100,000.

Central Personnel Data File (CPDF)

Two types of reporting made by agencies to the OPM include the Dynamic and Status files (quarterly and monthly, respectively) covering a range of employee personnel/payroll data.

Certain and Continuous Payment Option

A form of pension payment where the benefit is paid out for the lifetime of the participant with a specified number of payments guaranteed so that a beneficiary will receive payments until the end of the guarantee period if the employee dies before the guaranteed payments are complete. For example, under a ten-year certain and continuous payment option, a retiree who lives less than ten years receives payments until death, then the retiree's beneficiary continues to receive payments for the remainder of the ten year period. A retiree who lives longer than ten years continues receiving payments after the ten year period until death. Also knows as a "Term Certain and Continuous" payment option.

Certain Only Payment Option

A form of pension payment where the benefit is paid out entirely over a specified period of time—usually five, ten, or fifteen years—with no ongoing payments after the specified period. If the retiree dies before payment period is over, the remaining payments are made to a beneficiary. Also known as a "Term Certain" payment option.

Change To Lower Grade

- For positions under the General Schedule or under the same wage grade schedule, a change-to-lower grade changes the employee to a lower grade; and
- When both the old and new positions are under the same type ungraded wage schedule, or in different pay-method categories, a change-to-lower grade changes the employee to a position with a lower rate of basic pay.

Charge Out

A **Material Issue** used when the item is scheduled for future return.

ChartField

A field storing a chart of accounts, resources, and so on, depending on the PeopleSoft application. ChartField values represent individual account numbers, department codes, and so forth.

ChartField Balancing

PeopleSoft enables you to set up ChartFields and indicate that you want specific ChartFields to match (balance) on the debit and the credit side of a transaction. When you work with Controlled Budgets, the Fund and Budget Period are already set up in the system to balance (match). For example, suppose you want to balance by Class and Program. You indicate that these on a panel that these ChartFields are required, along with Fund and Budget Period which should already be selected. When you enter a transaction, you must enter the same Class, Program, Fund, and Budget Period ChartFields on both sides of the accounting entry. but you can modify any ChartFields, other than these four, on the user-defined line. The system always requires that total debits equal credits.

ChartField Combination Edit

Also called *Combo Edit.* The process of editing journal lines for valid ChartField combinations based on user-defined rules.

ChartKey

One or more fields that uniquely identify each row in a table. Some tables contain only one field as the key, while others require a combination.

ChartViews

Charts of data in the model, presented through the Worksheet which retains the ability to drag dimensions on the chart as desired.

Check In/ Check Out

The process of retrieving planning activities from the BAM database (check out) and posting changes and results back into the database (check in).

Child

A node or detail on a tree linked to another, higher-level node (referred to as the parent). Child nodes—projects, customers, and so on—can be rolled up into the parent. A node can be a child and a parent at the same time depending on its location within the tree.

Child

A node or detail of a tree linked to another, higher-level node referred to as the parent. Child nodes can be rolled up into their parent. A node can be a child and a parent at the same time depending on its location within the tree.

Chunking

Chunking is a PeopleSoft Enterprise Warehouse mechanism that makes voluminous processing easier through the use of multiple small parallel processes. By enabling chunking, multiple jobs are spawned from one Jobstream. These jobs run in parallel (behind the scenes) to process data efficiently.

Citizenship Code

Numeric indicator as to whether the employee is a U.S. citizen or a foreign national serving in the U.S. The codes are:

- citizen
- other

Civil Service Retirement System (CSRS)

A retirement plan available to employees of the federal government. CSRS covers all employees appointed to a position in the federal government before January 1, 1984. Coverage includes a basic annuity plan with employee contributions and the Medicare Hospital Insurance component (1.45%) of the Social Security tax.

Class catalog

Lists classes used to interface to an external system. A class contains data members of basic types and/or objects that are typed after another class. A Class can also contain lists of basic types or objects.

Class ChartField

A ChartField value that identifies a unique appropriation budget key when you combine it with a Fund, DeptID, and Program Code as well as a Budget Period. Formerly called “sub-classification.

Classification Code

Need App A code that categorizes an engineering change. Example classification codes include the following: Mandatory, Optional, Upgrade, Quality, and Safety.

Clock Hour Reporting

Method of reporting time by recording actual times in and out (start and stop) (see Time Reporting).

Clone

To create a unique copy of an object. When used in PeopleCode, clone will always mean to make a unique copy. Copy, on the other hand, may or may not mean making a unique copy. Copy may mean making a new reference to an object, so if the underlying object is changed, both the copy and the original change.

Cloning

The process that enables you to copy run controls to create employee schedules from existing Run Control ID's that have already been executed and saved.

Close Date

The date in which time entry is no longer allowed for a given pay period. Defined as an offset number of days to the pay period end date.

Close Price

The price of the final trade for a security at the end of the trading day.

Closure Calendar

A calendar that establishes closure dates for shipping, receiving, and materials management operations for a specific **Business Unit**. Typically, application processes account for these closure dates when determining Lead Time and dates for anticipated fulfillment processing dates (scheduled shipment dates, scheduled arrival dates, and lot retest dates, for example).

CMA (Census Metropolitan Area) Code

In PeopleSoft Workforce Analytics, the CMA code is prescribed by Statistics Canada, and refers to the main labor market area of an urbanized core with a population of at least 100,000.

COBRA (Consolidated Omnibus Budget Reconciliation Act)

In PeopleSoft Workforce Analytics, this refers to legislation that requires employers to offer continued health care coverage to employees, and their dependents, who lose benefits coverage under certain defined conditions such as voluntary termination, divorce, becoming an overage dependent, or retirement. Any individual, whether employee or dependent, that is covered under a health plan at the time of a qualifying event, has the option to elect COBRA coverage.

Codepage

One character set.

Collection

To make a set of documents available for searching in Verity, you must first create one or more collections. A collection is set of directories and files that allow search application users to use the Verity search engine to quickly find and display source documents matching various search criteria. A collection is a set of statistics and pointers to the source documents, stored

in a proprietary format on a file server. Since a collection can only store information for a single locale, PeopleSoft maintains a set of collocations (one per language code) for each search index object.

Combined Federal Campaign (CFC)

A vehicle used by federal employees to contribute to a charity or charities of their choice.

Commercial-Off-The-Shelf (COTS)

Equipment or software that is currently sold commercially to at least one customer.

Commission Tax Method

A payroll tax calculation method that adds year-to-date earnings to earnings for this pay period and finds the annualized gross by multiplying by the number of pay periods in the year; the gross is then divided by the number of tax periods specified on the paysheet. This method is used for Canadian processing only.

Commitment Control

Commitment control includes budget control and commitment accounting functionality.

Common Shares Issued and Outstanding

Represents the residual ownership interests in the corporation. This is the composite number of shares available and tradable on the open market.

Community Background

In the United Kingdom Community Background refers to the religious category, such as Catholic or Protestant, of employees, job applicants or appointees. See the Northern Ireland Report for more information.

Compa-Ratio

In PeopleSoft Workforce Analytics, Compa-Ratio is most commonly defined as the relationship between current pay and the midpoint calculated as: (Incumbent Pay/Midpoint)*100. Usually expressed in whole numbers, or in percentage form by dropping the multiplication operation. Much less common is the use of a compa-ratio calculation as: range midpoint/market rate.

Compensation Frequency

In PeopleSoft Workforce Analytics, this is the frequency at which a job is paid. This is the value you use for reporting or quoting pay. Examples include Annually, Monthly and Weekly.

Compensation Planning

In PeopleSoft Workforce Analytics, this is the process through which employee compensation plans are defined, and compensation budgets are allocated throughout an organization. Major components of compensation planning include designing pay structures, setting individual pay levels, and budgeting and forecasting compensation spending.

Compensation Rate

In PeopleSoft Workforce Analytics, this is the compensation rate for a job. This is the rate the company uses for quoting and reporting pay.

Comp time (compensatory time)

A PeopleSoft Time and Labor-managed employee benefit where time off is granted in exchange for time worked based on customer-defined criteria; is associated with an expiration and is used as reported time (see Attendance).

Compensation

The process by which a worker is remunerated for services rendered to, or work performed on behalf of a business entity.

Compensation Package

All of the base and non-base components on a job row.

Compensation Rules

Business methodology or logical process that is applied to reported time in order to determine payable time (see Time Administration).Competency

In PeopleSoft Workforce Analytics, Competency is a knowledge, ability, skill, accomplishment, or National Vocational Qualification (NVQ).

Competency Inventory

All of the roles, tasks, competencies and accomplishments possessed by the workers in the current workforce. This data is migrated from internal source systems into the data warehouse tables of the PeopleSoft Enterprise Warehouse.

Competency Strategy

The type and number of roles, tasks, competencies and accomplishments essential to accomplishing a business scenario based on your strategic business goals.

Competitive Appointment

An appointment to a position in the competitive service following open competitive examination or under direct-hire authority. The competitive examination, that is open to all applicants, may consist of a written test, an evaluation of an applicant's education and

experience, and/or an evaluation of other attributes necessary for successful performance in the position to be filled.

Competitive Service

All positions as defined by 5 USC 2102 in the executive branch of the Federal Government are in the competitive service unless they are specifically excluded from it. Positions in the legislative and judicial branches are outside of the competitive service unless they are specifically included.

Compress

The act of placing a Planning task as early as possible in the schedule without violating any constraints.

Compressed Split

In PeopleSoft Demand Planning, an optional function that allows a split database to be compressed so it can be transferred to an account manager's computer.

Concurrent Offerings

Multiple stock purchase offerings that are active and outstanding at the same time. The end date is measured from the employee's grant date.

Concurrent Processing

The situation in which you run multiple batch processes at a time. In PeopleSoft Benefits Administration, for example, simultaneous open enrollment and event maintenance qualifies as concurrent processing.

Configuration Code

A unique 50-character identification code that accurately tracks and costs inventory with the PeopleSoft Product Configurator. It corresponds to a lot number for a non-configured item.

Configuration Costing

The overall process of reviewing and evaluating anticipated cost data for a configured item.

Configuration parameter catalog

Used to configure an external system with PeopleSoft. For example, it might set up configuration and communication parameters for an external server.

Consolidate Assets

In PeopleSoft Asset Management, the process of consolidating multiple load lines, usually coming from a separate application, into one asset.

Consolidate Depreciation

In PeopleSoft Asset Management, the process of summing all open Add and Adj transactions by transaction type, **Transaction Date**, and accounting date for all composite members reporting to one composite asset.

Consolidated Bill

A grouping of bills gathered together for invoice presentation. The bills belonging to a consolidated bill are invoiced and printed together, with a page summarizing the bills as a group.

Consolidations

The PeopleSoft Pension Administration functions that accumulate hours, earnings, and pension contributions based on payroll data.

Consolidations-Elimination Set

A related group of intercompany accounts that is processed during consolidations. Once eliminated, this group of accounts should normally net to zero.

Constraint

A limit to a schedule, that, when violated, must be repaired to produce a valid schedule. User-configurable Planning constraints include Missed Request Dates, Missed Promise Dates, BI Shortages, RM Shortages, Capacity Overloads, Missed Inventory Targets, Changeovers, and Excess Inventory. See also **Temporal Constraint**.

Constraints

In the PeopleSoft Enterprise Warehouse, a constraint can consist of one or more filters and is used to define complex business logic. Constraints are based on DataMaps.

Consumption Pattern

In PeopleSoft Activity-Based Management, an attribute used to describe how an activity interacts with the cost objects to which it has been assigned. A unit type activity can expect to be performed on a regular basis so that each time a product is produced. A batch type activity may only be performed periodically for a given range of transactions. For example, each time a machine is setup to produce another product type. Sustaining type activities generally occur to support the overall operation of a company unrelated to products produced or customers served.

Contact

A person associated with a Customer ID. Contacts can be internal contacts or external contacts. Internal contacts are your employees who manage the relationship with your customers, from handling billing inquiries to product/warranty questions, to basic product/service questions. Interactions with customers can be recorded via PeopleSoft Conversations. Self service interactions can be recorded through PeopleSoft Contact Us. External contacts are your customer's representatives who can access self-service transactions

and receive documents such as sales order acknowledgements. Contacts must have a User ID to access self-service transactions.

Contact Us

A method by which customers and unregistered guest users send email messages to specific addresses or members of the merchant's organization. Merchants can also define automatic response messages.

Container

An Inventory stock unit for receiving, putaway, bin to bin transfers, picking, shipping, adjustments, and physical accounting. Each container is associated with a unique container ID.

Content Reference

Content references are pointers to some kind of content registered in the portal registry. These are typically either URLs or iScripts. Content references fall into three broad categories: target content, templates, and template pagelets.

Contextual reference

PeopleCode refers to a row or buffer field determined by the current context; that is, the context in which the PeopleCode program is currently executing.

Contingent Beneficiary

In PeopleSoft Pension Administration, any non-spouse pension beneficiary, including a child, other relative, or a trust. Spousal consent is required in order for an employee to name a contingent beneficiary.

Contracting Officer (CO)

Individual who has the authority and the official responsibility to produce a sound acquisition document.

Contracting Officer's Technical Representative (COTR)

Individual responsible for monitoring a contract and its associated tasks and deliverables.

Contractor

Any individual or non-employee reporting time that will not be paid through the payroll system.

Contribution

Represents money a stock purchase participant elects to contribute to the plan. Contributions are deducted from the participant's paycheck and used to purchase stock pursuant to the offering and purchase period they are enrolled.

Contributory Plan

A Pension plan to which employees contribute. Contributions are typically a percentage of pay deducted from the employees' paychecks.

Control Budget

Commitment control enables you to establish budgets that provide extensive, active budgetary controls over transactions, rather than just passively recording transactions.

Control ChartField

A control ChartField is a key ChartField that you designate to be the control field. Designating a ChartField as the control allows you to set attributes for a specific value of the ChartField that are different from the attributes specified for the budget type in general. For example, if the tolerance for a Projects budget type is set to 10% in general, you can override this value, making it higher or lower for specific projects.

Control Group

A mechanism to relate vouchers together for the purpose of controlling voucher input into PeopleSoft Payables. Generally used for assigning vouchers to data entry personnel and for reviewing input.

A set of parameters that determines the major forecast process options. The Control Group code is assigned to a group of **Forecast Item** and controls the forecast development and tracking for each item in the group.

Control groups are used by the Analytic Forecast Component to govern particular properties of the forecast rule, such as what accuracy to expect and what statistical method to apply. Forecast elements are assigned to exactly one control group. They manage differences among forecasts within a set.

Control Hierarchy

The relationship between business units, origins, vendors, and control groups in PeopleSoft Payables that defines which processing data will be automatically entered on each voucher.

Control Number

A sequential identifying number used to identify an exercise.

Control Plan

In PeopleSoft Quality, a plan that brings together application, measurement, and control and response criteria for a specific product and process.

Conversation

Any notes, transcript, or detail of a telephone call between an employee and a customer. Conversations may be tied to items, payments, purchase orders, document references, or bills of lading.

Conversion data profile

A conversion data profile takes the values from a particular PeopleSoft database table (such as the table holding bank transaction codes) and specifies how that value appears in PeopleSoft Business Documents.

Conversion data profile

A conversion data profile takes the values from a particular PeopleSoft database table (such as the table holding bank transaction codes) and specifies how that value appears in PeopleSoft Business Documents.

Conversion Loader

A sample SQR delivered with PeopleSoft Asset Management that transfers data from multiple fixed-length ASCII files into sample, relational conversion tables.

Copy Bill

In PeopleSoft Billing, the online environment providing for the replication of a single bill, generating a new bill with its own unique invoice number.

Core Functionality

Core functionality is the set of information in PeopleSoft HRMS that is common to your entire global workforce tracking needs—and is always displayed on the primary page.

Core hours

The hours a workday, workweek or pay period in which a time reporter must be present for work in a flexible work schedule (see Scheduling).

Corporate Account

In PeopleSoft applications, this is equivalent to the Account (ACCOUNT) ChartField. The term is used to make a distinction between the chart of accounts typically used to record and report financial information for management, stockholders, and the general public, as opposed to a chart of statutory (Alternate) accounts required by a regulatory authority for recording and reporting financial information.

Corporate Reporting

Companies with more than $10 million in assets whose securities are held by more than 500 owners must file annual and other periodic reports. Publicly held companies are required to file documents with the SEC which include:

- Registration statements for newly-offered securities
- Annual and quarterly filings (Forms 10-K and 10-Q)
- Proxy materials sent to shareholders before an annual meeting

- Annual reports to shareholders
- Documents concerning tender offers (a tender offer is an offer to buy a large number of shares of a corporation, usually at a premium above the current market price)
- Filings related to mergers and acquisitions

Corporate Repurchase

When a corporation elects to repurchase some of it's own securities. This reduces the Common Shares Issued and Outstanding. Typically, used to improve the valuation of the company's common securities outstanding as well as the Earnings Per Share (EPS).

Correction to IRR

An IRR type used when corrections need to be made to an original IRR that has already been submitted to the Office of Personnel Management (OPM). Federal employees covered by the CSRS retirement plan require SF-2806-1. Federal employees covered by the FERS retirement plan require SF-3101. A Correction IRR is also used if original retirement deductions were over-reported. See also Individual Retirement Record (IRR).

Correspondence Customer

A customer to whom all correspondence (statements) is addressed, often a corporate customer receiving correspondence for associated child customers.

Cost Accounting

A method where business costs are accumulated and distributed to products, processes, or discrete undertakings on an equitable basis. There are a variety of cost accounting methods, but they all share the same basic functions. classifying costs, recording costs, allocating costs to products or activities, summarizing and reporting costs to management. Cost accounting requirements and financial accounting requirements are not necessarily synonymous.

Cost Assignment

Resources assigned to cost objects or activities.

Cost Basis

Typically, this refers to the original price of an asset used in determining capital gains. However, in the case of death of an optionee, the appraised value of the asset at the time of death is the cost basis.

Cost Center

A Time and Labor Business Unit, in which all related costs attributable to some center within a business (such as an activity, an organization, or a program), are segregated for accounting or reimbursement purposes.

Cost Element

See **Inventory Cost Element** and **Manufacturing Cost Element**.

Cost Flow

Determines how depletions will occur for purposes of costing a transaction. Cost flows available include Specific Lot ID, Specific Serial ID, FIFO, and LIFO.

Cost Objects

Cost objects represent cost information about products, customers, and channels. They are the final results of the activities performed by your business, representing the focal point of costing and profitability analysis. Examples are products, customers and channels. They are the final results of the activities performed by your business. Your model's resources and activities are linked to the cost objects. They are often the focal point of profitability analysis.

Cost of Capital

An attribute used to describe the behavior of a particular cost object. A primary cost object is typically the main focus of the activity-based management analysis. This may be a product, customer or channel that you wish to calculate cost for. A support cost object may be used in a similar manner but may be further allocated to other support cost objects or primary cost objects.

Cost Of Living Allowance (COLA), Non-Foreign

A cost-of-living allowance payable to an employee at a location in a non-foreign area where living costs are substantially higher than those in the Washington, DC area.

Cost Profile

A combination of a receipt cost method, a cost flow, and a deplete cost method. A profile is associated with a cost book and determines how items in that book are valued, as well as how the material movement of the item is valued for the book.

Cost Profile Group

A grouping of items for the purpose of costing transactions and valuing inventory for a given book. Assigning an item to a cost profile group determines the books used by the item when accounting for that item.

Cost Roll-up

A process for calculating item costs. Cost roll-up provides a summation of all of the costs associated with the bill of material structure and the routing used in producing the item.

Cost Row

A cost transaction and amount for a set of ChartFields.

Cost Template

A collection of cost components that you can apply to a group of purchased items.

Cost Type

A user-defined method of categorizing item costs in Manufacturing for simulations and what-if analysis. Examples of cost types include current costs (which reflect the item's current bill of material or routing), proposed costs (which could be used in preparation for the next standard cost period), or activity-based costs (which include costs for items that consume a given activity).

Cost Version Type

A combination of cost types and cost versions used in cost rollups. Valid values include production (rolls up only manufacturing data and uses only the primary BOM and routing, each with a code of 1), engineering (can roll up with either manufacturing or engineering data, with any combination of BOM/routing codes), or simulation (only rolls up with manufacturing data, but can use any combination of BOM/routing codes).

Count Grade

A user-defined evaluation of a counting event.

Count Point

A predefined step on a routing or operation list where you can gather operation completion information. You define the appropriate points on the routing, record completions at these count points, and the system automatically backflushes the prior operations. This is only used on production IDs.

Counts

Count elements allow you to count the number of days or hours from a specific period of time. Counts are used primarily during proration calculations, but can potentially be utilized in other situations as well.

Court-Ordered Benefits Coverage

As prescribed in Title 5, United States Code and Title 5, Code of Federal Regulations, court orders that stipulate that an employee must continue or begin the coverage features for all employee benefits must be enforced. Federal employees are mandated by court orders to continue covering or begin covering their former spouses and/or children under their federal employee benefit programs (health, life, and thrift savings).

Court-Ordered Garnishments

As prescribed in Title 5, United States Code and Title 5, Code of Federal Regulations, court orders enforcing child support, alimony, or collection of commercial indebtedness are served on the appropriate entity within the Federal agency and implemented as offsets against the employee's salary.

Coverage

An employee's chosen benefit plan and coverage level; that is, what sort of benefit is provided as well as the value.

CPAM (Caisse Primaire d'Assurance Maladie)

In France, CPAMs are the local social security offices that manage health coverage for French workers. CPAMs are regulated and established by the French government. If you're managing a French workforce you'll need to identify and track the CPAM offices that impact your enterprise.

CRAM (Caisse Régionale d'Assurance Maladie)

In France, CRAM is the regional social security body which oversees the running of CPAMs. CRAM offices work with companies to both prevent and compensate workers for industrial injury.

Create Date

The date that you extracted a deduction or offset to PeopleSoft Deduction Management or created a split deduction.

Create Missing Items

In PeopleSoft Demand Planning and Inventory Planning, a feature that enables automatic system generation of master records that don't exist in the system.

Created Time

Time collecting device time or elapsed time generated by the system based on the time reporter's schedule (see Time Administration)

Creating Time

The preliminary generation of time segments as close as possible to their likely values when you officially report time—so that the information on the time records is as fresh and current as possible. The system shows you time that has already been created, rather than you having to create it "on the fly" when you come in to report. The process fills in reporting day gaps as defined by work schedules.

Credit Analyst

A required field used in PeopleSoft Receivables, Billing, Order Management, and Deduction Management when working with items. Each item must be assigned to a credit analyst. If no credit analyst is assigned to an item, the credit analyst assigned to the customer is used as the default.

Credit Risk Spreads

In the financial services industry, the additional charge to a risk-free interest rate, based on a riskier credit rating.

Credits

See Flexible Credits.

CREF

Acronym for Content Reference.

Crew Reporting

A Time and Labor process that enables you to report the earnings which consist of one or several time reporting codes and associated quantities of hours, amounts, or units, and task information for one date under report for a user-defined crew. The system transforms the information into instances of daily time for each crew member for the entered date.

Critical Success Factors (CSFs)

In PeopleSoft Balanced Scorecard, things that an organization must do well or excel at to achieve its goals. One or more key factors or objectives that must be accomplished for a particular strategic thrust. Key Performance Indicators are attached to CSFs.

CRM Warehouse

See Warehouses.

Cross Border Walker

This term is used in Europe for an employee who lives near a border in one country and works in another country. Such employees are subject to different tax and social security rules.

Cross-Plan Validation

The process by which the PeopleSoft Benefits Administration determines enrollment prerequisites for benefit plans. You can define four types of cross-plan validation prerequisites: prerequisites based on plan types, benefit plans, dependent enrollments, and coverage percentage limits for Life and AD/D plans.

Cross-View Reconciliation

In PeopleSoft Demand Planning, a process that enables the balancing of forecasts between selected levels of related views with the same **Forecast Item** key. The process is used when adjustments have been made to a working view and are then required in a related view.

Cube

See **Multidimensional Database** (MDDB).

Cube View

In PeopleSoft Demand Planning, defines the user's own view of a forecast. The parent working view and dimensions determine what forecast data is included and how aggregates are formed.

Cumulative Tax Method

A payroll tax calculation method that adds together year-to-date earnings and earnings for the current pay period, then annualizes the result before calculating tax. This method is useful when Payrolls vary greatly in amounts from pay period to pay period, such as in the case of sales commissions.

Currency Calendar

In the financial services industry, business calendars for markets outside the organization's domestic operations that reflect the foreign markets' holiday schedules.

Currency Conversion Engine

A PeopleSoft Enterprise Warehouse Engine that processes financial information in multiple currencies.

Current Period

The earliest pay period for which the close date has not passed (see Time Reporting).

Current Period (Time and Labor)

In Time and Labor, the employee's current pay period which will be determined via the employee's Pay Group affiliation. Although there can be only one definition of Current Period per installation, the user can change it manually.

Current View

A reporting screen in Time and Labor whose effective date is within the date boundaries of an employee's current pay period, and for which pay has not yet been confirmed. A *Future Time Reporting Transaction* is one that has an effective date after the last day of the employee's current pay period. An *Historical Time Reporting Transaction* is one that has an effective date before the first day of the employee's current pay period.

Current Year

A period for event maintenance processing.

Curve Generator

A supporting module (common to financial services industry applications) that enables you to construct curves used to determine appropriate interest rates for given maturities and / or time periods. You can import market data from outside sources such as Bloomberg, upload the data from a spreadsheet, or manually enter the data. You can then build configured curves from segments or combinations of other curves.

CUSIP Number

A nine digit alphanumeric number associated with issuers' securities. CUSIP (Committee on Uniform Securities Identification Procedures). A uniform numbering system widely used to identify specific securities and their issuers.

Custom Statement

A user-created logical or mathematical expression that determines information about an employee in PeopleSoft Pension Administration. Custom Statements commonly define employee groups and benefit formulas.

Customer Inquiry

A window containing options to review customer balances, aging, history, items, actions, and conversations.

Customer Scorecard

See PeopleSoft Customer Scorecard.

Customer Tree

A user-defined graphical representation of your current sales organization. A customer tree is used to establish and distribute funds and to determine authority levels for promotional activities.

Cut Session

Cut sessions are a means of dividing a course session. You use cut sessions where a course session does not run on consecutive days from start to finish, or if there are multiple instructors or locations. Each cut session has it's own start/end date, location, and instructor. For example, if you have a course that runs for two days a week for a month, you would divide the course session into four cut sessions, each of which is two days long.

Cycle Count

A manual counting event that does not cover an entire inventory business unit. Usually includes every item (and lot, if applicable) in a location or family.

Cycle Interval

The number of days between cycle counts.

Cycle Procedures

Inventory planning tasks that need to be performed on a regular basis to ensure an up-to-date **Inventory Policy**. The tasks can be performed either at the end of a processing period or within the period, and should always be performed if the forecast or **Control Group** or **Policy Item** parameters change. Tasks include generating a policy and reviewing **Work Queue** messages.

D

DAT file

A text file (input.dat) used with the Verity search engine that contains all of the information from documents that will be searchable but not returned in the results list.

Data Elements

Data elements, at their most simple level, define a subset of data and the rules by which to group it.

For PeopleSoft Balanced Scorecard, data elements are used as the basis for key performance indicators, and as target values for Key Performance Indicator (KPI) objects.

For Workforce Analytics, data elements are rules that tell the system what measures to retrieve about your workforce groups.

Data Entry Access List

Used to present a concise list of often-performed data entry tasks to a user. You can assign multiple control plans to a single data entry access list.

Data Extract

A report that creates a file used to transmit data to a third party on magnetic media. There is no meaningful printed output for this type of report.

Data Loader

Data Loader is a PeopleSoft Enterprise Warehouse utility that moves data from the Operational Data Store staging area to either the ODS reporting area or the Data Warehouse. The Data Loader utility is made up of several pages that allow you to enter Metadata to define your source and target records and your transformation rules and then perform the load by running an Application Engine.

Data Loader Map

Defines how to extract data from the Operational Data Store (ODS), transform it, and load to a Target Table. The target table can reside in the warehouse or the ODS layer.

Data Manager

A PeopleSoft Enterprise Warehouse engine that distributes revenue, expense, analytical application engine results, statistical quantities and other measures across business units, departments, products, customers and channels—any field or logical group in the chart of accounts. You can define a number of types and options within this engine. It is also used as a means of posting to the Performance Ledger.

Data Manager Rules

In the PeopleSoft Enterprise Warehouse, Data Manager rules use Constraints to specify the source as well as the target tables for moving, aggregating, or multidimensionalizing your engine output. Rules use Data Manager methods to enrich the PeopleSoft Enterprise Warehouse data.

See Data Manager Methods.

Data Manager Methods

There are several methods: Copy, GL Mapper, Prorata, Spread Even, and Tree Aggregation. Each method enables you to move and/or enrich engine output.

Data Mart

A Data Mart is a data structure that uses a central fact table and related dimension tables to generate a "relational cube" or directly generate an Insight report.

Data Mart Builder

The Data Mart Builder is a multiple Application Engine (AE) process, that is, a framework of procedural programs, that creates a Data Mart.

DataMaps

Information that builds upon the data captured in the TableMap records. DataMaps enable you to define a logical view of the physical PeopleSoft Enterprise Warehouse tables. DataMaps bring together information from many different tables and fields and define it all as one entity or table.

Data Row

Contains the entries for each field in a table. To identify each data row uniquely, the system uses a key consisting of one or more fields in the table.

DataSet

DataSets are used as input for various engines and processes, for instance, the Analytic Forecasting component, the Data Manager, user defined functions, drivers in Activity-Based Management, and data elements in the Key Performance Indicator Manager. DataSets provide a user defined set of information to the engines. DataSets use Constraints to restrict used columns and restrict returned rows. Each DataSet is created by a process specific setup. However, the underlying logic is the same, enabling you to more easily understand the functional aspects of the process.

Data Warehouse

A large database containing data summarized from one or more transactional systems, optimized to support the analysis needs of the enterprise. An ideal data warehouse contains all the data necessary to make business decisions. Users analyze the data in the warehouse using Online Analytical Processing (OLAP) tools and ad hoc query/reporting tools. An increasing

number of organizations have "virtual" data warehouses, where the data warehouse is not one physical database, but rather a collection of specialized (and distributed) data marts.

See also PeopleSoft Enterprise Warehouse.

Data Warehouse Tables

Data Warehouse tables act as the portal for getting data into the PeopleSoft Enterprise Warehouse from PeopleSoft, OLTP applications or other “outside” sources. These tables are used:

- As targets for loading operational data.
- For error detection and handling
- For data validation.
- For aggregation.

Database Alias

The PeopleSoft Pension Administration utility that looks up employee data.

Dataset

A file containing data to be analyzed by the Quality Server program. The dataset is similar in content to a spreadsheet.
In PeopleSoft Planning, a file that stores schedule information such as tasks, resources, calendars, and so on.

Date

See **Accounting Date Transaction Date** or **Effective Date**.

Date

If you want to either include a date in a calculation, or determine a new date by taking a starting date and either adding or subtracting a period of time to come up with another date, you use a date element.

Date Classified

Date the Position Description is approved by Management/Position Management.

Date Eligible To Retire

Date an employee is eligible to optionally retire based on the combination of age and service that meets legal requirements.

Date Under Report

The date (day) in PeopleSoft Time and Labor for which time is being reported. The Date Under Report does not have to equal today's date.

Day Breaker

Customer defined time that is used to determine when one day becomes the next. It's used to determine the "logical" date of a punch. (See Understanding Workgroups.)

Days Supply

In PeopleSoft Inventory Planning, a method that can be used with several types of **Inventory Policy**. Using this method, a specific number of days of supply for an item should be used to calculate the item's inventory policy.

Deal Type

PeopleSoft Treasury has categorized deals into several basic deal types from which you can choose when defining an instrument.

Death Coverage

The PeopleSoft Pension Administration function that determines the factor used to reduce an employee's benefit when the plan charges for PRSA coverage.

Decompressed Split

In PeopleSoft Demand Planning, a function for returning a compressed split database to its original form. See also Compressed Split.

Deduction

Any amount taken from an employee's pay check each pay period. Deductions may include health or medical benefits, union dues, and so on. See also Benefit Deduction and General Deduction.

Deduction Date

The as of date for the deduction item in PeopleSoft Receivables.

Deduction Item

An individual item that you created in receivables and is an open receivable on the customers account due to a deduction that they took in a payment for a receivable item.

Deduction Reason

Code that describes the type of deduction. When assigned to a write-off resolution, it determines what accounting entries to create.

Deduction Specialist

The individual responsible for tracking and resolving deductions in PeopleSoft Deduction Management.

Deduction Subset

A group of deductions selected from a company's standard set of deductions. Deduction subsets minimize data entry time in special processing situations such as bonus check runs.

Default Mode (DM) model

In the financial services industry, an approach used by financial institutions to predict a decline in portfolio value. Only two outcomes are considered – default or non-default. If the debt does not default, there is no change in the value. If the debt does default, then the loss is calculated as the difference between what was contractually owed and the value of any collateral recovered.

Defection Analysis

In PeopleSoft Workforce Analytics, the identification of employees who are likely to leave the organization based on predefined assessment criteria.

Deferred Compensation

Compensation payments that are payable to an individual in the future such as pension plan payments, annuities, stock awards and profit sharing. Note: Profit sharing can be considered direct pay if paid out in cash on a periodic basis or deferred pay if cumulative with the intention of payment in the long-term future.

Deferred Vesting

The adjustment made to the original option's vesting schedule that pushes the vesting into the future.

Defined Benefit Plan (DB Plan)

A retirement income plan (usually called a pension plan) where the employee's benefit is definitely determinable based on a plan-specified benefit formula.

Definition or Function Definition

The parameters for any of PeopleSoft Pension Administration's nineteen core functions. A definition has to be explicitly associated with an employee Group Definition before it can be applied.

Dekit

The ability to return material issued in kits to inventory. This is used when entire kits need to be returned; individual components are handled through kit issues/returns.

Delete Non-Matching Items

In PeopleSoft Inventory Planning, an option used in the Generation process to delete Inventory Planning items that don't have corresponding items in Demand Planning. The item deletion occurs when the system generates the policy.

Delta

When retroactive processing occurs for a given payee, the system recalculates each element generated for the payee. The system compares the recalculated results to the original results. The difference between these results is typically referred to as the retro "delta." A retro delta can represent either an underpayment or an overpayment that results in an adjustment to the payee's earnings.

Demand

Collection of training requests. This could be an employee demand, a departmental one or a company-wide demand.

Demand Filter Width

In PeopleSoft Demand Planning, specifies the confidence interval within which demand is considered to be reasonable. Actual demand that is outside the confidence interval is automatically filtered and replaced by the value at the edge of the interval. The value is expressed as a percentage.

Demand Filtering

In PeopleSoft Demand Planning, provides a way to detect and highlight unusual demands and forecast errors. If the demand falls outside of a band that is considered reasonable, the system automatically adjusts it to the level of the boundary and logs a message to the **Work Queue**.

Demand Number

The configured product sub-component sequence number.

Demand Planning

In PeopleSoft Activity-Based Management, this type of planning focuses on studying the impact of cost objects and activity volumes.

Demand Priority

The placing of importance on independent demand. The Planning engine uses the demand priority value to determine the order in which you fulfill the demand. You can set a demand priority from 1 to 998 with 1 being the most important level. The priority value of 999 is reserved for the system.

Demand Priority Rules

In PeopleSoft Inventory, a set of rules that will sort demand so the most important demand will have the first opportunity to reserve available inventory. If demand priority rules have

been defined, the Material Reservations process (INPLDMND) sequences orders by priority rank, processing those with the lowest rank value first.

Deplete Cost Method

Determines how you cost a depletions transaction within a book. The deplete cost methods available include Actual, Non-Cost, Perpetual Weighted Average, Periodic Weighted Average, and Value at Current Standard.

Depreciate When in Service

A switch that indicates whether PeopleSoft Asset Management should allocate depreciation as of the date an asset was placed in service. This is valid only in the year the asset was acquired.

Depreciation - Declining Balance

Budgeting calculates this as: Cost minus Accumulated Depreciation divided by Life divided by number of periods per year. It results in a higher depreciation expense in the early years of an asset, which decreases as you near the end of its useful life.

Depreciation - Double Declining Balance

Budgeting calculates this as: Cost minus Accumulated Depreciation multiplied by 2 divided by Life divided by number of periods per year. It results in a higher depreciation expense in the early years of an asset, which decreases as you near the end of its useful life.

Depreciation Methods

The various methods of spreading the aquisition cost across the life of an asset rather than expense the full value of an asset at the time you acquire it The value of the asset consequently decreases (or depreciates) through time. The four depreciation formulas delivered with PeopleSoft Budgeting include: declining balance, double declining balance, straight line, and sum of the years.

Depreciation - Straight Line

A method of depreciating asset value in equal amounts across the life of the asset. Per-Period Straight-Line depreciation is calculated as the cost of an item minus the salvage value divided by the number of periods to depreciate.

Depreciation - Sum of the Years

A depreciation method equal to the value of the remaining years of life divided by the sum of the years remaining is multiplied by the Net Book Value. This figure is then multiplied by the percent of years to depreciate. This results in a higher depreciation expense in the early years of an asset, which decreases as you near the end of its useful life.

Depromote

In PeopleSoft Demand Planning, the process of making an adjustment to actual demand data that removes the effect of a promotion during a defined period. As a promoted period moves into history, the system creates an adjusted demand entry that is equal to the **Prorated Forecast**.

DeptID

A ChartField that defines departments or administrative offices that have operational, fiscal and/or budgetary responsibility for specific sets of activities.

Derived Metric

The result of a calculation on a report of base metrics.

Detail

A temporary assignment to a different position for a specified period when the employee is expected to return to his/her regular duties at the end of the assignment. This employee is considered for pay and strength count purposes to be permanently occupying his/her regular position. Unless the agency chooses to use an SF50, a detail is documented with an SF52.

Detail Tree

A tree that employs ranges of detail values under each node; you must manually specify the detail values.

DFI ID (Depository Financial Institution ID)

A PeopleSoft Payables bank identifier, consisting of Transit Number, Swift ID, or CHIPS ID.

Dimension

A single element of a business model, such as product, department, or location. Cube Manager uses the term Conforming Dimension.

In terms of data analysis, dimensions can be thought of as criteria, such as time, product, and location, used to pinpoint a particular piece of data. For example, in the retail industry a set of dimensions could be geography, product, time, customer, and vendor. The geography dimension would include company, chain, region, district, and finally store attributes. A dimension is also a column heading on an analysis and reporting template which you can drill through or roll up to the multiple levels.

In PeopleSoft Budgeting, a view option that assists in summarizing the rows of data in line-item budgeting.

A single element of a budgeting model, such as account, product, project, department, or operating unit. In PeopleSoft Budgeting, these dimensions typically represent the ChartFields used by your organization during your budgeting process.

Dimension Table

In the PeopleSoft Enterprise Warehouse, Dimension Tables store additional attributes or data about Facts. Some example dimensions include Customer, Channel, Geography and Product.

Direct Compensation

In PeopleSoft Workforce Analytics, Direct Compensation is payment made to workers in exchange for their contributions to the organization. Direct Compensation is typically categorized as including Cash Compensation and Long-term Variable Compensation.

Cash payments made to workers in exchange for their contributions to the organization. Direct pay is typically categorized as fixed pay (for example, base pay, shift differentials) and variable pay (for example, profit sharing, incentive, bonus). Note: Profit sharing can be considered direct pay if paid out in cash on a periodic basis or deferred pay if cumulative with the intention of payment in the long-term future.

Direct Calculation

Calculate actual and directly assigned dollars.

Direct Cost

In PeopleSoft Workforce Analytics, a direct cost of an activity or a cost object. An example is the salary cost of employees working on a project.

Director

An affiliate of the company who holds a seat on the Board of Directors for the corporation. A Director, generally, is not an employee of the corporation.

Disability and Discrimination Act of 1995

In the United Kingdom this act makes it unlawful to discriminate against individuals on the basis of their disability in relation to recruitment, promotion, training, benefits, terms and conditions of employment, and dismissal.

Disability Rate Code

The desired percentage of disabled persons that should be employed by French employers, as mandated by the French government.

Disbursement View

In PeopleSoft Demand Planning, a **Forecast View** that allows the forecast from a working view to be reported on using an alternate key. Disbursement views are built directly from the working views and inherit many working view attributes, including time period and associated user data definitions, from the parent working view.

Discounted Stock Option

Rights to a stock option at a price less than 100 percent of fair market value at the time of grant.

Discretionary Plan

In PeopleSoft Workforce Analytics, this is a plan for distributing compensation awards that provide managers the ultimate discretion over a pool of money which is either funded based on company, group, or employee performance, or it's budgeted. The discretionary award determination is sometimes guided by a pre-determined percent of the participant's salary, expressed as an opportunity. This figure can then be modified based upon management's perception of actual value created by the group or employee.

Disqualifying Disposition (DD)

When an optionee sells or otherwise disposes of the shares of stock acquired through the exercise of an incentive stock option or through an employee stock purchase plan before the holding period for preferential tax treatment has lapsed.

In the case of Incentive Stock Options, the holding period is one year of the date of exercise and two years of the date of grant. At the time of disposition, the individual recognizes compensation income equal to the difference, if any, between the option price and the fair market value of the corporation's stock on the date of exercise. If the sale price is less than the fair market value of the stock on the date of exercise, the compensation income is limited to the total sales price less the total option price, less any fees.

In the case of purchases through an employee stock purchase plan, the holding period is one year from the purchase date and two years from the enrollment date. Compensation income in a disqualifying disposition is equal to the difference between the total fair market value on the purchase date and the total purchase price.

Distribution

Provide a repository of time and associated estimated and actual allocated labor costs to other systems

The process of assigning values to ChartFields. A distribution is a string of ChartField values assigned to items, payments, and budget amounts.

Distribution Network

A distribution network is a prioritized list of Inventory business units (IBUs). When a customer orders a product, the system uses this network to determine which warehouse the stock ships from.

Distribution Profile

A definition of ChartField distributions assigned for compensation costs. A distribution profile can be used to set up defaults for how the system should distribute costs associated with a position's salary, benefits, and earnings. PeopleSoft Budgeting-specific.

Distribution Rule

You use distribution rules to determine the order in which the system searches for matches against the distribution sets matrix when sales orders are entered.

Distribution Set

Distribution Sets assign account distribution information to combinations of defining elements used on sales orders.

Distribution Type

An identifier that defines one of the different transactions that move an item into or out of an inventory business unit. Distribution types are used to create debit and credit transactions to the general ledger via the Journal Generator.

Dividend

Distribution of earnings back to shareholders, prorated by the class of security and paid typically in the form of money or stock. The amount of a dividend is decided by the Board of Directors and is usually paid quarterly.

Document Management

The process through which a user has complete control of document version including the ability to view, query, and edit documents in a secure vault. Document management enables you to seamlessly perform online document queries and view documents directly, launching them from within PeopleSoft applications. You can associate pertinent documents with engineering change requests (ECR), engineering change orders (ECO), item revisions, bills of material, manufacturing and engineering routings, production component lists, and production operation lists.

Document Sequence Number

A value that the PeopleSoft system assigns to a document (such as an invoice, voucher, or journal) when you create a document for a business unit that you have enabled for document sequencing. The system determines the number by the values of the business unit, accounting date, and document type.

Document Sequencing

A flexible method that sequentially numbers the financial transactions (for example, bills, purchase orders, invoices, and payments) in your system for the purpose of statutory reporting and tracking of commercial transaction activity. Document sequencing requires that you classify all financial transactions into three transaction types—journal type, journal code, and document type—and that within each transaction type, all documents you enter are numbered sequentially. When you create a document (such as an invoice, voucher, or journal), the PeopleSoft system assigns a document sequence number to that document.

Document Type

The final level of three categories for defining a financial transaction (or document), necessary when using document sequencing. It represents the business purpose of a financial transaction, such as domestic customer invoice or customer credit memo. Document type is within one and only one journal code; journal code is within one and only one journal type. Document type is the only required category, because the values of the other two categories can be derived from document type.

Dollar Tolerance

In PeopleSoft Inventory, the acceptable cost difference between expected cycle count quantities and actual quantities counted. This value allows a margin of error for an item during cycle count reconciliation based on item cost.

Domestic Relations Order (DRO)

A preliminary version of a court order (usually stemming from a divorce settlement) ordering a division of a participant's pension benefits. The order is not in effect until it is determined to be "qualified" by virtue of meeting certain requirement. At that point it becomes a Qualified Domestic Relations Order, or QDRO.

Double Byte Characters

If you're working with Japanese or other Asian employees, you can enter the employee's name using double-byte characters. The standard double byte character set name format in PeopleSoft applications is: [last name] space [first name].

Draft Worksheet

A work space used in PeopleSoft Receivables to track a draft through its processing life cycle.

Drill-Back Calculation

Assigns indirect dollars and Drill-Back calculations. Also, this picks-up all costs in the Calculations Detail (CALC_DETAIL_F00) that was assigned during direct calculations.

Drill Down

The ability to go down to the next level of detail in a set of data. For instance, if you're looking at an expense figure for a division, you can drill down to the expenses for each department in the division.

Drill Down

The ability to go down to the next level of detail in a set of data. For instance, if you're looking at an expense figure for a division, you can drill down to the expenses for each department in the division.

Driver Lookup Table

Tables associated with a driver that enable different rates and amounts unique to a budget center.

Drivers

In PeopleSoft Activity Based Management, drivers are a means of assigning dollar amounts from resources, activities, and cost objects to other resources, activities, and cost objects throughout the model in PeopleSoft Activity-Based Management. Drivers can also be assigned across business units. There are different types of driver categories (transactional, duration, and intensity), and different ways of specifying how those dollar amounts are calculated (amount, percentage, spread even, and direct), as well as different ways that dollar amounts are assigned (depending on assignment type and object type).

In PeopleSoft Business Planning, a driver can be defined as a set of values that are used as an input to another process. In this context, a driver should be interpreted as a projection of external factors and other indicators. The user can the define a relationship between the driver and a financial result. The driver values and the defined relationship then combine to produce a projection of the financial result. For example, a projection of the number of square feet used (driver) and the price per square foot paid in rent (driver) can combine to produce a projection of rent expense (financial result).

DRO

See Domestic Relations Order.

DSS (Decision Support System)

A DSS is a workstation-based analysis and reporting system, typically aimed at analysts and line managers. OLAP tools provide a powerful DSS.

Duration

In PeopleSoft Pension Administration, the utility that calculates the length of time between two dates.

Duration [Global Payroll]

An element type that calculates a period of time between two dates. For example, if you want to determine a payee's age, you can calculate the duration between his birth date and the calendar period end date.

Dynamic Group

A group in Time and Labor that enables you to establish criteria or attributes for a group of employees. All employees who fit this criteria at processing time belong to the group.

Dynamic Tree

A tree that takes its detail values—*Dynamic Details*—directly from a table in the database, rather than from a range of values entered by the user.

Dynamic Views

In PeopleSoft Demand Planning, a **Forecast View** that allows interaction with the forecast data using an alternate key structure. By using dynamic views, you streamline the working view and can complete the working-view design without having to anticipate all conceivable adjustments.

E

Earliest Change Date

Determines both the range of dates and the amount of data that will be processed for each time reporter (see Batch Processing)

Early Punch

A punch that is more than the predefined number of hours/minutes before a scheduled punch where a time reporter is warned

Early Retirement Date (ERD)

A retirement date earlier than a plan-specified “normal” retirement date. Employees usually must meet age and/or service requirements to be eligible for early retirement, and early retirement benefits are often reduced to compensate for the longer duration of payments.

Early Retirement Factor

The reduction made to an employee's benefit if the employee elects for early retirement.

Early Warning

In commitment control, warning of possible future budget exceptions. You can specify that you are to receive a warning when commitments and expenditures reach a predetermined percentage of budget. For example, you can instruct the system to let you know when commitments and expenditures reach 50%, 80%, or some other percent of a budget.

Early/Late Adjustments

The PeopleSoft Pension Administration function that calculates early retirement factors or late retirement factors.

Earning Group

Part of a group of defaults assigned to job codes. Earnings group may include non-salaried items such as holidays and bonus pay dependent on individual company parameters.

Earnings

The amount owed to an employee based on salary, hours worked, or other calculation routines, plus other types of compensation and holiday, vacation, and bonus pay.

Earnings [Global Payroll]

An element type that defines the different types of compensation that are added to a person's pay. Examples include salary, commission, bonuses, and retirement pay.

Earnings Accrual Class

Categorizes a set of accruable earnings.

Earnings Code

Codes that represent the various types of earnings such as regular, overtime or leave.

Earnings Per Share (EPS)

The portion of a company's profit allocated to each outstanding share of common stock. Net income (reported or estimated) for a period of time is divided by the total number of shares outstanding during that period.

Earnings Type

An abbreviated and encrypted set of business instructions containing compensation instructions. Earnings Type may also contain Benefit Entitlement and Administration instructions, taxation instructions, Financial Accounting instructions, Organizational Administration instructions, work group and labor affiliation instructions, and other instructions.

Economic Loss

In Funds Transfer Pricing, this refers to the break fund economic loss, calculated by applying the theoretical value of the interest rate differential (IRD) against a cash flow stream, based on the amount of the prepayment or cancelled draw-down.

Economic Value Added

In the financial services industry, Economic Value Added is a financial metric that factors into the measurement of an activity's profitability the cost of economic capital assigned to that activity.

EDGAR (Electronic Data Gathering, Analysis, and Retrieval)

An electronic system implemented by the SEC that enables companies to file documents in conjunction with disclosure requirements mandated by the SEC.

EDI Agent

Used in EDI processing, the inbound EDI Agent loads trading partner data (flat files) into the PeopleSoft database using transaction, map, and trading partner definitions set up using EDI Manager. The outbound EDI Agent extracts information from the PeopleSoft database and generates data files that can then be processed for transmission to a trading partner.

EDI Manager

A suite of online pages used to define transaction sets, trading partner profiles, and translation maps for EDI transactions.

Edit Table

A table on the database that has its own record definition, such as the Department table. As fields are entered into a PeopleSoft application, they can be validated against an edit table to ensure data integrity throughout the system.

EEO Company Code

In the United States companies are assigned this federal code for EE0 and VETS100 reporting.

Effective Date

A method of dating information in your system. You can predate information to add historical data to your system, or postdate information in order to enter it before it actually goes into effect.

Effective Date

A method of dating information in your system. You can predate information to add historical data to your system, or postdate information in order to enter it before it actually goes into effect. The Effective Date usually defaults to your system's current date.

Effective Periods

In PeopleSoft Demand Planning, the number of periods of historical demand used in the Model Reset process. The value can be used to exclude older, possibly unrepresentative historical demand data from model **Optimization**.

Effective Sequence

A system-generated number assigned to distinguish between two job entries with the same effective date.

Effective Tax Rate

The ratio of income tax paid over gross income, showing the percentage of income actually paid in taxes.

Effectivity Date

The date on which a component can be added or substituted in production, typically specified on an item's bill of material.

EIS (Executive Information System)

An EIS is a workstation-based analysis and reporting system for executives. An EIS provides a higher-level view of the data than a DSS, and typically requires less knowledge about the underlying transactional systems. OLAP tools provide a powerful EIS.

Elapsed Schedule

A method of scheduling a time reporter's time that is based on TRC and duration. This method can be used for scheduling of elapsed time reporters (see Scheduling.)

Elapsed Time

Reporting non-clock time in increments of hours or partial hours (see Managing Time / Understanding Time Reporting).

Elapsed Time Service

A method of calculating a period of service that uses only the start and end dates of the period to determine the amount of service. Hours worked or other measures of the actual work performed during the period are not taken into account.

Electronic Certification System (ECS)

An automated Payment Voucher authorized by the Certifying Officer for use within the Treasury Department, Financial Management Service's financial system. PeopleSoft provides a method to record and generate data files for on- and off-cycle processed payments.

Element

In PeopleSoft Global Payroll, an element refers to both primary elements and supporting elements. Primary elements are comprised of earnings, deductions, absence entitlements, and absence take elements. Supporting elements are element components that are combined to create primary elements.

In PeopleSoft Enterprise Performance Management, elements are used to create a Profile. An Element can be one or more columns of data in an Enterprise Warehouse table, associated with a single dimension (for example, Customer, Product, Department, or Channel). An Element can also be KPI, Population, subscription data from a third party, preexisting Profiles, and data mining scores.

Element Group

Element Group identifies a group of elements to provide eligibility. You can then use this as a notational shortcut—instead of having to list each element, you can use the element group name. Element Group's expedite the process of manipulating earnings and deductions.

Element Name

Name assigned by the user for data fields, rules, formulas, and tables. For example, the names you give to new rules, elements, or objects.

Element Segment

When an element changes mid-period, requiring the affected element (and perhaps a subset of other elements) to be calculated multiple times on either side of the date on which the change takes place, element segmentation is used. Unlike period segmentation, the system segments only the elements you select, and creates separate result columns only for the specified elements. In element segmentation, there is only one gross-to-net result set.

Eligibility Group

Eligibility groups define the possible earnings, deduction, absence entitlement, and absence take elements that a payee might be eligible to receive. This enables you to group payees so as to assign eligibility for certain pay elements.

Eligibility Rule

PeopleSoft Benefits Administration uses eligibility rules during Benefits Administration processing to determine which benefit programs and options an employee is eligible for. Eligibility rules are closely associated with event rules: they determine what options an employee can *have,* while event rules determine which of those options an employee will actually be able to *choose.*

Elimination Set

See **Consolidations-Elimination Set**.

Email Template

Pre-defined parameters that establish automatic email generation during budget submittal, rejection, publishing, and/or targeting.

Employee

An individual employed by an organization and administered as an employee in the PeopleSoft Human Resources system.

Employee Accounts

The PeopleSoft Pension Administration function that tracks employee contributions to a pension plan.

Employee ID

A unique identification code for an individual associated with your organization.

Employee Paid Benefit

The portion of a pension benefit funded by the employee's own contributions to the pension plan. Also, the PeopleSoft Pension Administration function that determines this amount.

Employee Profile

This PeopleSoft Activity-Based Management feature enables time and labor information to be part of an analysis.

Employee Stock Purchase Plan

A type of statutory stock option plan through which employers grant options to their employees in order to provide them with additional forms of compensation.

Employee Survey

In PeopleSoft Workforce Analytics, a method for capturing information about the activities performed by a given employee as well as the amount of time they spend performing each activity to perform activity-based management.

Employee Survey Report

The Employee Survey Report is an annual regulatory report that the French government requires from employers with more than 200 employees. In French it is called "Le Bilan Social". The report is communicated to both labor unions and the government. It provides a snapshot view of the company over the past 3 years for about 200 indicators.

Employee Training Cost

Amount budgeted to pay for students' salaries while on training courses.

Employer Identification Number (EIN)

In the United States a company is typically defined as a business enterprise that has a unique federal Employer Identification Number (EIN) for payroll tax reporting purposes.

Employer's Liability Insurance Associations (Berufsgenossenschaften)

Social Insurance in Germany is maintained and administered by private organizations that act as employer's liability insurance associations. Employers pay out premiums to these associations, who administer and pay out funds to workers who are injured on the job.

Employment Cost Index (ECI) Adjustment

Annual increase to wages established/permitted by statute.

Employment Equity Computerized Reporting System (EECRS)

Canadian companies are required to report to the Federal Government on employment equity. PeopleSoft Human Resources contains the Canadian Employment Equity report (PER101CN), which creates a data interface file to the federal government's Employment Equity Computerized Reporting System (EECRS).

Employment Record Number (EMPL RCD#)

A field in PeopleSoft Human Resources Management Systems and PeopleSoft Workforce Analytics that indicates an employee has multiple job records in the system. A numeric value (0, 1, 2) is assigned to each job as a way to uniquely identify that job record.

Encumbrance

A claim against funds. It is a projection of future expenses based on the situation, as we know it today. Encumbering funds is not the same as spending them or even guaranteeing that you will spend them. It just means that if the situation as it exists today does not change, you will spend all of those funds by the end of the fiscal year.

Engineering Bill of Material (EBOM)

A listing of all the parts, raw materials, and subassemblies that form the basis of all item and product structures. EBOMs differ from MBOMs (Manufacturing Bills of Material) in that they are not visible within Production Planning or Production Management and are isolated from Manufacturing.

Engineering Change Order (ECO)

A revision to a blueprint or design, released by engineering to modify or correct a part and/or bill of material. PeopleSoft Engineering uses ECOs to manage and document required assembly and component changes.

Engineering Change Request (ECR)

A document that allows you to request manufacturing process improvements and report product defects directly to the engineering department. When workflow is enabled, ECRs can also be routed for review and approval, after which they change into ECOs.

Engineering Cost Version

The process of generating cost versions for new and modified configurations based on engineering bills of material (EBOM) and costing data.

Engineering Workbench

An engineering environment, separate from production, consisting of engineering bills of material (EBOM), engineering change requests, engineering change orders, EBOM cost roll-up capability, online BOM comparisons, and seamless integration to a document management vault.

Engineering Workbench

An engineering environment, separate from production, consisting of engineering bills of material (EBOM), engineering routings, engineering change requests, engineering change orders, EBOM cost roll-up capability, online BOM comparisons, and seamless integration to a document management vault.

Enterprise

In PeopleSoft Time and Labor, all of the business units of the installation site.

Enterprise Performance Management (EPM)

See PeopleSoft Enterprise Performance Management

Enterprise Portal

The PeopleSoft Enterprise Portal is a separate product offering purchased independently of any other PeopleSoft applications. It can be used with or without any PeopleSoft application. It can be used as a standalone corporate portal that does not access PeopleSoft data at all.

Enterprise Resource Planning (ERP)

The encompassing term for all the transaction-oriented database applications an organization deploys across it's business enterprise. The term includes financial, manufacturing and supply chain, human resources, and payroll applications, among others.

Enterprise Warehouse (EW)

See PeopleSoft Enterprise Warehouse

Entry Authority

Authorization granted by employees to specific user IDs for entering expense data on their behalf.

Entry Currency

The currency used to enter budget data.

Entry Event

An automated process that generates multiple debits and credits resulting from single transactions, to produce standard supplemental accounting entries.

Entry Event Code

Designation of an Entry Event; an identifier or label.

Entry Event Generator

A mechanism that generates standard, supplemental accounting entries based on Entry Event codes.

Entry Event Process

An accounting transaction. Entry Event processes combine to form Entry Events. For example, requisition posting is a Purchasing process, and cash clearing is a Payables process. Each process can involve one or several Entry Event Steps.

Entry Event Step

Part of an accounting transaction. For example, the BUDG process includes these steps, among others: prepare allotment budgets, prepare organization budgets, and prepare revenue estimates. Entry Event steps combine to form Entry Event processes.

Entry On Duty Date (EOD)

Date that indicates when an employee started to work at his/her current agency.

Entry Type

Any activity that creates or updates an item.

EPM (Enterprise Performance Management)

See PeopleSoft Enterprise Performance Management

Equal Employment Opportunity Commission (EEOC)

In the United States the EEOC requires that most companies file one or more reports from a series named EEO-1 through EEO-9. These reports include counts by federal employment categories of male and female employees in certain ethnic groups.

Equitization

A process that enables parent companies to calculate the net income of subsidiaries on a monthly basis and adjust that amount to increase the investment amount and equity income amount before performing consolidations. For organizations with complicated parent/subsidiary business unit relationships, this automated process reduces the process time and reduces the possibility for errors.

Equity Increase

In PeopleSoft Workforce Analytics, Equity Increases are base pay increases granted to bring an employee's pay up to some internally specified standard for your organization.

Equivalent Standard Deviation

In PeopleSoft Demand Planning, a **Standard Deviation** developed during the Model Reset process that enables you to compare standard deviations from different models. The deviation is calculated by multiplying the **Model Equivalency Factors** defined on the **Control Group** by the standard deviation.

ERISA (Employee Retirement Income Security Act of 1974)

The U.S. Federal legislation enacted to prevent abuses of employee pension rights by employers.

Error Exception

A transaction that is stopped because the budget limits would be exceeded if it continued. For the transaction to proceed, action must be taken, such as canceling or reducing the transaction amount, increasing the budget amount, overriding the budget limits, or transferring available funds from another budget.

Error Ratio

The ratio of the statistical Standard Deviation to the base component that gives an indication of the accuracy of the forecast. The ratio is presented in the PeopleSoft Demand Planning Audit and Accuracy Review and is calculated during the **Forecast Calculation Process**.

Estimated Gross

Estimated labor cost associated with reported time (see Managing Time, Understanding time Reporting Codes).

Estimated Shipments

A group of shipment schedules used to manage sales order requested shipment data and actual shipment data. Used in conjunction with weight and volume pricing and freight charge calculations.

Ethnic Code

The Federal Office of Management and Budget (OMB) racial and ethnic census categories used for classifying individuals in U.S. Government reports.

ETL (Extract-Transform-Load)

See Extract-Transform-Load.

ETL maps

ETL maps provide rules for importing your source data to the data warehouse tables.

Evaluated Receipts Settlement (ERS)

A PeopleSoft Payables feature that matches receipts against purchase orders and generates vouchers without requiring an invoice.

Evaluations Periods

In PeopleSoft Demand Planning, indicates the number of future periods to use for the calculation of forecast errors. For example, if the number of evaluation periods is two, then the forecast error in April 2001 (after posting demand for April) would be determined by comparing the actual demand for April and the April forecast generated in February 2001. Evaluation periods are set separately for each view.

Event

Events are predefined points either in the Application Processor flow or in the program flow. As each point is encountered, the event fires on each component, triggering any PeopleCode program associated with that component and that event. Examples of events are FieldChange, SavePreChange, OnRouteSubscription, and so on.

Event

Occurrence or happening.

Event Class

An event or type of event that results in a change of benefits eligibility for an employee or dependent. Event classes are prominently used in COBRA and Benefits Administration processing.

Event Maintenance

The process that enables you to manage ongoing enrollments during a plan year. Changes involving maintenance include new hires and re-hires, terminations, family status changes, and changes to benefits eligibility.

Event Rule

Used by PeopleSoft Benefits Administration to determine how events are processed by the system. Event rules look at the benefit plan options an employee is eligible for and determine which options the employee can actually *choose.* Event rules are closely associated with eligibility rules but it is important to note that they are not the same. Event rules *should not* be used to determine eligibility.

Event Trigger

You use triggers to tell the system that when a change takes place to certain data (an event), it should perform an action automatically. When the event occurs, the system writes a line to a trigger table. Then when it's time for the action, the system reads the data from the trigger table and performs the appropriate action.

EW (Enterprise Warehouse)

See PeopleSoft Enterprise Warehouse.

See also PeopleSoft Enterprise Performance Management (EPM).

Expected Losses

In the financial services industry, the amount the institution predicts it will lose in portfolio value. Loan loss reserves are set aside to cover the expected losses.

Excepted Service

As defined by 5 USC 2103, the Excepted Service consists of those civil service positions that are not in the competitive service or Senior Executive Service.

Exception

User or system delivered, defined conditions applied to scheduled, reported or payable time that require audit or review (see Time Management)

Exception Rules

A rule (s) that is applied to scheduled, reported time, and payable time in order to determine conditions which require audit or review (see Time Administration).

Exception Severity

The degree of importance associated with an exception. For example, in exception which is a result of an employee clocking in late may have a Medium severity, while an exception which is a result of an employee not clocking in has a High severity (see Time Management).

Exception Time Reporting

A method of time reporting where only differences to the schedule are provided (see Time Reporting).

Excess Plan

A pension plan where the benefit formula provides an increased benefit for Final Average Earnings above a specified integration level. This compensates for the fact that Social Security benefits are based only on earnings up to a specified maximum.

Exchange Rate Variance

In PeopleSoft Cost Management, the change in currency exchange rate between the time the item is received into inventory and vouchered in Accounts Payable.

In PeopleSoft Payables, a matching feature that compares the exchange rate on the purchase order and the invoice and then copies any variance to PeopleSoft Inventory tables for analysis and accounting purposes.

Exclusive Pricing

Supersedes all pricing structures in effect for customers and products, except **Buying Agreement**, and enables you to drive pricing with a promotional structure. Exclusive pricing can be set up for a specific time frame and associated with particular orders.

Executive Schedule (EX)

Compensation and pay plan used by the Executive Branch of the federal government. Statutory pay limits are derived from several of the pay levels within this plan and imposed on the General Schedule and other existing pay plans throughout the Federal government.

Exercisable

The option shares that are available to the optionee to exercise.

Exercise

The transaction in which an individual purchases or "exercises" the right to purchase the option shares. The IRS refers to the purchase of company stock in an employee stock purchase plan as an exercise.

Exercise Date

The date on which an individual purchases underlying shares from and option grant or transacts a simultaneous purchase and sale of underlying option shares through a cashless exercise and collects option profit in cash or shares.

Exercise Price

The price per share required to exercise a stock option.

Exercise Proceeds

Cash, stock or other recognition received by a company as a result of option exercises, including cash or stock paid by individuals to exercise options and cash company tax savings from deducting non-statutory option profits at exercise.

Expense Location

Geographic area defined to enable the recording, tracking, and reporting of expense activity.

Expense Location Amount

Authorized spending defined for an expense type in a particular expense location and currency.

Expense Location Group

Collection of expense locations based on a common classification such as state, country, or continent.

Expense Type

Means of itemizing various kinds of business expenses. Examples are hotel, dinner, or ground transportation.

Expense Type Edit

User-defined requirement that mandates input of additional data—such as an airline ticket number or number of nights in a hotel—when an expense type is selected in an expense report.

Expense Type Group

Expense types that are classified together for reporting and tracking.

Expensed Item

Non-inventory item which may represent software, manuals, documentation, or any item for which no quantity on hand is maintained, but which can be specified on a bill of material (BOM). Expensed items can only exist as components on a BOM and cannot have a BOM, routing, or production area/item definition.

Expiration

The process by which the outstanding shares of an option cease to be exercisable, generally at the end of the option term. The length of the option term and the date of expiration are established in the Grant Agreement.

Expiration Date

In PeopleSoft Inventory, the date a lot exceeds its Shelf Life and is no longer acceptable for fulfillment or consumption. (Expiration Date = Creation Date + Shelf Life)

Expiration Date

The last day of an option term in which the option is canceled and no longer exercisable.

Expiration Grace Period

When you enter a stock action allows the exercise of the already vested shares as of the action date, the system will calculate the date these shares expire based on the grace period defined on the Stock Action Rules page for that stock action. The system will automatically cancel vested shares not exercised at the end of the expiration grace period.

Express Customer

A customer for whom the minimum necessary information is entered.

Expressions

Expressions enable you to create pseudo-columns made up of mathematical calculations based on actual fields on a table. Since expressions are resolved at run-time, duplicate information is not stored on the database.

Express Order

An order entry shortcut in PeopleSoft eStore and Mobile Order Management whereby the customer populates the shopping cart and goes directly to the order summary to checkout, bypassing any billing or shipment modification screens. Billing and shipping information defaults in as previously entered.

External Data

Data from external sources. For instance, in PeopleSoft Workforce Analytics, external data may include third party salary surveys and benchmark metric surveys.

External Scheme

In the United Kingdom an External Scheme is a vocational training, education and job placement program involving an employee, an employer and the government.

External System

Any system that is not directly compiled with the PeopleTools servers.

Extra Time

Any hours worked outside of an employee's normal (scheduled/shift) hours or days. Extra time may be scheduled in advance of when it is worked, and may be subject to special compensation rules. It may be treated differently than standard time for purposes of Benefit Entitlement and Administration.

Extraction

A reusable query that specifies what information should be retrieved from the Quality database.

Extract-Transform-Load (ETL)

The extraction and transport of data from one server to another remote server. In PeopleSoft budgeting ETL specifically refers to the process by which financial and human resource data is extracted from PeopleSoft Financials and HRMS and transferred to the PeopleSoft Enterprise Warehouse which PeopleSoft Budgeting uses to access and record data transactions. Within PeopleSoft Enterprise Warehouse, data migration typically refers to information moved from outside sources into the Operational Data Store tables.

Extrinsic Rewards

Tangible rewards that can be given to the individual. Typically categorized as financial and non-financial rewards. Financial rewards would include direct compensation, indirect compensation and deferred compensation. Non-financial rewards are provided to the individually and viewed as a benefit by the individual based on the culture of the organization such as the size or location of one’s office.

In PeopleSoft Workforce Analytics, tangible rewards given to an individual. Typically categorized as financial and non-financial rewards. Financial rewards would include direct compensation, indirect compensation and deferred compensation. Non-financial rewards are provided to the individually and viewed as a benefit by the individual based on the culture of the organization such as the size or location of one’s office.

F

Fact

Facts are numeric data values from fields from a source database as well as an analytic application. A fact can be anything you want to measure your business by, for example, revenue, actual, budget data, or sales numbers. A fact is stored on a fact table.

Fact Table

A fact table is where facts are stored in the PeopleSoft Enterprise Warehouse.

Fair Labor Standards Act (FLSA)

A federal regulation governing several time and labor issues. *FLSA Overtime* requires that all nonexempt employees be paid at a rate of time-and-one-half for all hours over 40 physically worked during a workweek. This requirement may be superseded by state or local laws when the lesser law is to the greater benefit of the employee, or by union contract. An *FSLA Workweek* is a permanently established, regular workweek for a group of employees.

Fair Market Value (FMV)

The price of a company stock based on the current market value as determined by supply and demand, or a valuation method. The stock market sets the fair market value for a public company. For a private company the fair market value is more subjective, but typically determined by financial factors or set by an outside valuation company.

Fair Market Value Tracking Methods

Methods used to track and report trading activity on various exchanges (i.e. NYSE, AMEX, NASDAQ, etc…).

Family Medical Leave Act (FMLA)

A federal regulation that protects health benefits and job restoration for employees who must take a leave from work to care for themselves or family members. FMLA regulations contain provisions regarding employer coverage, employee eligibility and entitlement, notice and certification, continuation of health benefits, and job restoration. PeopleSoft Benefits applications offer FMLA Plans that help employers and employees determine FMLA eligibility and schedule and track FMLA leave requests.

Federal Employee Group Life Insurance Program (FEGLI)

Generally, if the employee has Federal retirement coverage or is on a temporary appointment exceeding one year, he/she is eligible to participate in the FEGLI program. Once eligible, he/she is covered automatically for Basic Life Insurance and premiums will be deducted from gross salary unless coverage is waived within the first period of eligibility. The program offers Basic Insurance coverage and three types of optional coverage: Option A (Standard), Option B (Additional), and Option C (Family).

Federal Employee Pay Comparability Act (FEPCA)

This law provides a structure and methodology to determine and authorize locality-based pay adjustments to Federal employees in order to elevate their basic pay to be commensurate with private sector employees working in the same occupations in the same geographic localities. It also includes a feature to authorize agencies to make advance salary payments to attract candidates for open positions which have consistently been hard-to-fill in certain geographic areas.

Federal Employees' Compensation Act (FECA)

This law provides compensation and medical benefits to civilian employees of the United States for disability due to personal injury or disease sustained while in the performance of duty. A feature of this law provides for the continuation of pay (COP) without charge to leave for up to 45 calendar days due to disability and/or medical treatment following a traumatic injury. Employees file claims with the U.S. Department of Labor, Office of Worker's Compensation, which adjudicates the claims and compensates the employing agencies for the employee's pay and benefits during the claim period.

Federal Employees Health Benefits (FEHB)

Generally, the employee is entitled to coverage by the FEHB program if appointed to a position with Federal retirement coverage or has been on the rolls on a temporary appointment for more than one year. The Federal employer shares the cost of the premium (about 75%); actual premiums depend on the plan selected. If under a temporary appointment, the employee pays both the employer and employee shares. If the position is part-time, the employee pays the employee share and a portion of the employer’s share.

Federal Employees Retirement System (FERS)

A retirement plan available to employees of the federal government. FERS covers all employees appointed to a position in the federal government after January 1, 1987. Coverage includes Social Security, a basic annuity plan, and a TSP.

Federal Employer Identification Number (EIN)

Used to identify the tax accounts of businesses. Businesses, which have employees or operate business as a partnership or corporation, must obtain an EIN.

Federal Insurance Compensation Act (FICA)

Employee and employer contributions to Social Security.

Federal Reserve Transit Number

A unique identifier for U.S.-based banks, allowing banks to transfer funds within the Federal Reserve system.

Feeder Line

A type of production line replenishment used in PeopleSoft Flow Production. If you are using feeder line replenishment, smaller production lines create subassemblies that feed directly to your production line.

FEGLI Living Benefits Act

Beginning 7/25/95, a Federal employee who is terminally ill may elect to receive a lump-sum payment equal to the full amount of basic life insurance only, or a limited portion designated in multiples of $1000. An election to receive this benefit is irrevocable; the individual is considered terminally ill if his /her life expectancy is 9 months or less.

FICA (Federal Insurance Contributions Act)

FICA consists of both a Social Security (retirement) payroll tax and a Medicare (hospital insurance) tax. The tax is levied on employers, employees, and certain self-employed individuals.

Fictitious Calculations

Fictitious calculation rules perform temporary calculations. A fictitious calculation is a sub-calculation run during a normal calculation to determine a net that would have been computed if certain parameters were used. This result is then used for further processing in the normal calculation. A fictitious calculation is always started from inside a normal calculation, run for one payee, and run for a specified set of periods.

FIFO (First In First Out)

Method used by companies to record Disqualifying Disposition Income. If a company uses this method they record the optionees disposition of shares by attributing the shares to the earliest exercise, purchase or release dates for which shares remain available for sale.

Fill-In Employment

Employment held by persons during the time period after leaving their regular occupation in anticipation of, but before entering, military service.

Filter

A filter creates a subset of information. Filters are used in templates to limit your information from a pick list of attribute values.

Final Average Earnings (FAE)

The PeopleSoft Pension Administration function that averages earnings from a specified period of an employee's career. The result is used as a component of the pension benefit formula.

Final Forecast

The final forecast is the prorated version of the adjusted forecast, summarized to all levels of the product hierarchy. This is the best-guess version of the forecast that is used to make all decisions dependent on the forecast.

Final Table Merge Engine

Final Table Merge Engine is used by the PeopleSoft Enterprise Warehouse; it moves enriched data from one table into another. When you run a job in a jobstream, the immediate results are stored in temporary tables. At the end of the jobstream, the Merge Engine runs and merges the output temporary tables into the final tables, where processing can continue.

Financial Accounting

The accounting for a business entity's assets, liabilities, revenues, and expenses to determine its net worth and to produce financial statements. Within Generally Accepted Accounting Principles, a business has some latitude as to when and how to record its financial transactions, as long as it continues to meet its legal and regulatory requirements. A business' financial accounting requirements are not necessarily the same as its cost accounting requirements. The one should not be mistaken for the other (i.e. the extent to which a company's financial accounting system meets its cost accounting needs depends on how it has chosen to describe its chart of accounts and the level at which it has chosen to record financial transactions.

Financial Instrument

In the financial services industry, a specific product or service sold by a financial institution to its customers. In terms of the reporting hierarchy, a product falls under a ledger account, while an instrument falls under a product. A product may be treated like a generic description or category, while an instrument is a specific instance of a category.

Financial Performance Measures (FPM)

For the financial services industry, the Financial Performance Measures program performs calculations on financial instruments based on the rules defined in the Financial Calculation Rules module, and using input from the Cash Flow Generator, Stratification engine, Product Pricing, and Curve Generator. Its calculations include: measures of duration, option-adjusted spread and option-adjusted cost for PeopleSoft Funds Transfer Pricing, and Monte Carlo simulation for PeopleSoft Asset Liability Management.

Financial Product

In the financial services industry, a product or service sold by a financial institution to its customers. In terms of the reporting hierarchy, a product falls under a ledger account, while an instrument falls under a product. A product may be treated like a generic description or category, while an instrument is a specific instance of a category.

Financial Services Instrument

In the financial services industry, products created by financial institutions and sold to retail customers. Product prices and interest rates are set by the financial institutions and take into account its customers' behavioral models.

Financial Statement Simulation

A facility within Planning & Simulation which establishes rules for simulating future period, or pro-forma, financial statements. The user defines corporate financial policies, such as corporate tax rates, dividend distribution frequency, and force balancing rules, which are then applied to cash flows for a given future accounting period. The Financial Statement Simulator engine drives costs and revenues to accounts on PF_LEDGER_F00 via a scenario.

Financials Warehouse

See Warehouses.

First Year Amount

See 1st Year Amount.

Fixed Basis

The basis option enables you to create the data for the Basis online, as part of the rule. Fixed Basis is used with the Allocation Manager only. It is available with all methods except when Period-Based Allocation is being used. The Fixed Basis is a predetermined table that can be populated online.

Fixed Offering

The offering type is fixed when the end date of each offering is the same for all employees regardless of the employee's grant dates.

Fixed Percentage

A fixed percentage value. The source pool amount will be split based on this percentage to get the target amount. Used with the Allocation Manager.

Fixed Period Requirements

In PeopleSoft Enterprise Planning and Production Planning, a lot-sizing technique that sets the order quantity to the demand for a given length of time.

Fixed Picking Bin

A dedicated picking location for an inventory item. Fixed picking bins are replenished from bulk locations when the available quantity falls below the optimal quantity.

Fixed Plan

A stock purchase offering period where the ending offering date will be the same as the purchase date. Eligible employees will always purchase stock on the specific purchase dates and by the purchase rules you define.

Fixed Quantity

An **Inventory Policy** method that defines a fixed amount of an item to be ordered to meet replenishment needs. This method can be selected as an inventory policy for order quantity, safety stock, **Reorder Point**, and minimum and maximum parameters.

Fixed Source

The fixed source option enables you to create the data for the Source online, as part of the rule. Fixed Source is used with the Allocation Manager only. It is available with all methods except when Period-Based Allocation is being used. The Fixed Source is a predetermined table that can be populated online.

Flexible Credit

Any credit associated with a given benefits program, plan, or type of coverage. Credits based on an entire program can be applied toward the benefit costs however the employee chooses.

Flexible Hours

Hours during the workday, workweek or pay period during which a time reporter covered by a flexible work schedule may choose to vary his times of arrival and departure from the worksite (see Scheduling)

Flexible Spending Account (FSA)

An account to which an employee and (optionally) an employer pledge an annual amount for a plan year. The employee then submits claims for authorized expenses.

Flexible TimeSpan

A user-defined period into which costs can be collected. Flexible TimeSpans can be as long or as short as you like—covering multiple years or a single day. The main purpose of Flexible TimeSpans is to assist you in analyzing costs.

Flexible Work Schedule

A method of scheduling a time reporter's time that is based on a range of flex hours of start and stop times and core work hours. This method can be used for scheduling clock and elapsed time reporters (see Scheduling)

FLSA Status

A PeopleSoft Human Resources term that is used to indicate whether a job is exempt or nonexempt according to the Fair Labor Standards Act. All employees associated with a

particular job will receive that job's FLSA Status. FLSA Status is an eligibility determination factor for PeopleSoft Benefits Administration.

Forecast Attribution

A FSI (financial services industry) transformation process through which forecasted product originations are pooled and run through the cash flow engine for future periods.

Forecast Calculation Process

In PeopleSoft Demand Planning, the process by which a **Statistical Forecast** is generated for each item at each level of the view. When a **Forecast Item** is set to recalculate, the system tries several forecast calculation methods and picks the one with the least amount of error. This process also makes adjustments for promotions and filters for abnormal demand.

Forecast Definition

Forecast definitions are a set of forecasting rules that generally govern multiple forecasts distinguished by key properties such as products, customers, channels, and so forth.

Forecast Element

Each forecast within a single definition is called a Forecast Element.

Forecast Fulfillment

In PeopleSoft Demand Planning, a process used to manage forecasted demand over a period of time. The process makes it possible to divide the total forecast demand into portions so that certain portions can be met, even if the total forecast cannot be met entirely.

Forecast Item

In PeopleSoft Demand Planning, a logical item used as the basis to forecast demand. The components of a forecast item key are defined for each level in a forecast view.

Forecast Level

See Level.

Forecast Period

A period in time as defined by the calendar for which data is processed through the PeopleSoft Demand Planning model.

Forecast Start Period/Year

Determines the most recent period for which demand data is available for a forecast view. This period can also be described as the last actual demand period to have had an impact on the forecast.

Forecast View

See **View**.

Foreign Education

Education acquired outside of any state of the U.S., the District of Columbia, the Commonwealth of Puerto Rico, a Trust Territory of the Pacific Islands, or any territory or possession of the U.S.

Form 10-K

A form used for annual reports pursuant to Section 13 or 15(d) of the Securities Exchange Act of 1934 for which no other form is prescribed.

Form 10-Q

A form used for quarterly reports under Section 13 or 15(d) of the Securities Exchange Act of 1934, filed pursuant to Rules 13a-13 or Rule 15d-13. This report, which public companies are required to file quarterly with the SEC, provides unaudited financial information and other selected material.

Form 5500 Participant Count Extract

A PeopleSoft Pension Administration data extract containing data that a plan administrator needs in order to complete IRS Form 5500, used to report on the number of plan participants.

Form S-8

A form used to register securities offered by a reporting company under its employee benefits plans, including stock option plans. Also called the Registration Statement under the Securities Act of 1933.

Form W-2

A form used by employers to provide workers with a statement of wages, tips and other compensation from the previous year. This form, distributed employees by January 31 of each year, reflects state and federal taxes, social security, Medicare wages, and tips withheld.

Formula

Element which enables you to define your own formulas for use—gives further flexibility to define complex organizational needs.

Formula Plan

This compensation distribution plan type is based on a pay out rule, as the pay out rule is defined. The pay out rule can be based on a flat amount, a percentage, or a data element. Whereas a Target Plan distributes pay out based on a comparison of a performance measure against a target, in a Formula Plan the pay out is based just on the pay out rule.

French Professional Elections

French companies employing a certain number of employees must hold elections for selecting personnel representatives (Délégués du personnel), and members of the Work Council (Comité d'Enterprise).

Frozen Rate

A rate that is applied to allocate resources to activities in place of the actual, budgeted and capacity rates calculated by the Activity-Based Management (ABM) Engine.

FTE (Full Time Equivalency)

FTE is the percent of full time the employee should normally work in this job. Full time is defined by the Standard Hours specified in either the Salary Plan Table or the Default Standard Hours specified in the Installation Table.

FTP (Funds Transfer Pricing) Adjustments

Adjustments made to the PeopleSoft Funds Transfer Pricing (FTP) base rate, for such factors as geographic premiums, liquidity premiums, embedded options, or incentive programs.

FTP (Funds Transfer Pricing) Base Rate

In PeopleSoft Funds Transfer Pricing (FTP), this refers to the basic charge or credit that is applied to a ledger account, a product, or an off-balance sheet position.

FTP (Funds Transfer Pricing)

See PeopleSoft Funds Transfer Pricing.

Full-Time Equivalent

See FTE.

Function

A category of pension calculation. PeopleSoft Pension Administration divides a pension calculation into nineteen "core functions" such as Service, Final Average Earnings, and Benefit Formula.

Function Result

The calculation rules for any of PeopleSoft Pension Administration's nineteen core functions. These rules match Definitions—the specific parameters for the function—to the Groups of employees that use that particular definition. Function Result also refers to the value produced by the rules.

Fund ID

In the financial services industry, Fund ID is a lookup code used to track investment funds associated with a financial instrument or account. Provided primarily by the financial analytic applications to track investment funds for insurance policies.

Funds Transfer Pricing (FTP)

See PeopleSoft Funds Transfer Pricing.

Fungible

This term describes a resource used for multiple activities.

Future Period

Any pay period which is not current and whose close date hasn't passed (see Time Reporting).

Future Periods

The number of periods of future forecasts maintained by the PeopleSoft Demand Planning system.

G

Gang Reporting

See Crew Reporting.

General Deduction

Any non-benefit deduction. Examples include charitable deductions, union dues, parking, garnishments, and bonds. General Deductions are calculated from the General Deduction Table; Benefit Deductions draw on one of the benefits tables.

General Ledger Distribution

The process and guidelines by which accounting information is transferred from your PeopleSoft Receivables or Deduction Management system to a general ledger system.

General Schedule (GS)

Compensation and pay plan used by the Executive Branch of the federal government.

Generation Control

Generation control elements allow you to indicate to the system whether to process an element based upon criteria you define. There are six parameters that control this function and comprise the definition of the generation control element—HR Status, HR Action/Reason, Segment Status, Frequency, Formula, and Run Types.

Generic Conversion Factor

A conversion factor that applies universally between two units of measure. The factor is used in the conversions between levels of PeopleSoft Demand Planning **Forecast Item**s and Inventory Planning **Policy Item**.

Generic Process Type

This term applies to Process Scheduler. Process types are identified by a generic process type. For example, the generic process type "SQR" includes all SQR process types, such as "SQR Process," "SQR Report," and so on.

Geo RSZ Code

This code is for Belgian employers to track the geographical location for RSZ codes.

Geographic Location Code

In Canada this code is prescribed by the government and refers to the location a business is in.

Giveaway Adjustment Type

In PeopleSoft Order Management, the price break tables are set up to indicate what product the user receives as a free premium based on defined quantities or prices. The giveaway item does not have to be the same product that the customer is purchasing. For instance, you may set up a price break that indicates that a purchase of 100 widgets entitles the customer to one or more free T-shirts. The system automatically adds an order line for the free item. Giveaways cannot be applied to the total order.

Goals Matrix

In PeopleSoft Workforce Analytics, a matrix used to create calculation rules for group or employee performance goals. You can combine and standardize multiple performance goals into a single, weighted, goal score, against which actual performance is compared. A Goals Matrix can be used to in conjunction with a pay out distribution plan called a Target Plan.

Grace Period

A period that is a number of hours or minutes before or after a scheduled punch where a time reporter’s punch is accepted. For Stock Administration, the period of time an optionee has to exercise an option after termination and before the option expires.

Grade

A range of pay in a graduated scale that includes positions of different occupational groups. The work performed should be equivalent as to the level of difficulty and responsibility and the level of qualification requirements of the work. The levels are established and designated within a specific pay plan by law or regulation.

Graduate Education

Successfully completed education in a graduate program for which a bachelor's or higher degree is normally required for admission. To be creditable, such education must show evidence of progress through a set curriculum, i.e., it is part of a program leading to a master's or higher degree, and not education consisting of undergraduate and/or continuing education courses that do not lead to an advanced degree.

Grandfathered Benefit

A benefit that an employee was entitled to prior to a change in the plan and that defines the employee's new minimum level of benefits. The change might be caused by a plan merger, new legislation, or a plan amendment.

Grant

A contractual right giving an individual the option to purchase a specified number of shares of stock through an Equity Compensation Plan. Also known as an option.

Grant Agreement

The legal document issued by a company defining the number of shares granted, grant price, vesting schedule and other terms and conditions of the stock option or stock award.

Grant Date

The date the individual begins participating in a stock purchase offering. The date on which an option or other award is granted. The date the company enters into the grant agreement. The underlying stock's fair market value on this date generally derives the option price.

Grant Price

The price per share at which the stock option was granted. This is the price per share the individual must pay when exercising the option.

Gross Salary

The sum of an employee's salary and earnings defined as part of gross salary. The gross salary is used to calculate budget amounts for benefit plans defined as a percentage of an employee's salary.

Gross-up

The process used to calculate taxes and resultant gross pay from a check for an exact net amount.

Group

In PeopleSoft Billing, a specific term for a posting entity composed of one or more transactions (items, deposits, payments, transfers, matches, or write-offs).

Group

Any set of records associated under a single name or variable in order to run various calculations in PeopleSoft Business Processes. In Time and Labor, for example, employees are placed in groups for time reporting purposes, while in Administer Variable Compensation, groups identify which employees are eligible for what forms of compensation. In PeopleSoft Pension Administration, you'll use Custom Statements to define criteria for grouping employees, then by associating calculation rules (Definitions) with specific Groups, you can vary rules for different classes of employees.

Group Asset

A financial asset with no cost information. It is used to depreciate the sum of the costs of its associated group member assets.

Group Asset Depreciation

The depreciation of a group asset calculated using an average service life set by a local regulatory agency and a calculated group depreciation rate.

Group Coverage (Or Generic) Qualification Standards

Standards prescribed for groups of occupational series that have a common pattern of education, experience, and/or other requirements.

Group Member Asset

A financial asset with cost information. Cost information for all group members of a group asset is summed up to the group asset level, where depreciation is calculated.

Group Security

The ability to grant or deny access to groups. You can set up group security by Group ID or by user ID.

Group Security [Time and Labor]

The ability to grant access to employee time, by providing security through Time and Labor's groups functionality. For example, you might want your employees to only access their own records, or allow your supervisors who handle all of the time input for have access to specific groups. You can restrict the user from accessing everyone, or allow the user to be able to access only their own records, or only a specific group. This feature also provides the ability for employees to report their own time.

Group Type

An indicator of the activity that created the billing group: billing, maintenance, payment, transfers, or unposted.

H

Handicap Code

A code that identifies a type of physical or mental impairment that substantially limits one or more of an employee's major life activities.

Hazard/Disposal Code

An inventory item group sharing a disposal routine.

Headcount

The number of people represented by a given Employee Survey record in the PeopleSoft Enterprise Performance Management product line.

Health and Safety Executive (HSE)

Health and Safety reporting for your UK operations is sent to the local office of the HSE per the requirements of the RIDDOR (Reporting of Injuries, Diseases, and Dangerous Occurrences Regulations).

Health Benefits Code

An alpha/numeric code that identifies each Health Benefit plan.

Health Benefits Effective Date

Date the health benefit plan goes into effect or the effective date of cancellation.

Hierarchy

Hierarchy refers to the relationship between the levels in a dimension.

Highly Compensated Employee (HCE)

An IRS employee category applied to employees who are considered “highly compensated” according to a federally set standard. This distinction is used for the purposes of nondiscrimination tests, to determine that Section 401 and Section 129 plans do not discriminate in favor of highly compensated employees.

HIPAA

The Health Insurance Portability and Accountability Act of 1996. PeopleSoft Benefits applications enable you to comply with this act, which requires that employers provide Certificates of Group Health Plan Coverage to employees who have their health coverage terminated. This certificate lists group health coverage an employee had for the twelve month period prior to the date coverage ended as a result of termination of coverage. The HIPAA certificate will be used by subsequent health coverage carriers to evaluate pre-existing condition clauses, if applicable.

Historical Periods

In PeopleSoft Demand Planning, a component that indicates the maximum number of periods of historical demand maintained for a **Forecast Item** within a **Forecast View**. Historical periods must be a minimum of two years in order to support the development of seasonal models based on an item's demand history.

Historical Rules

An element used to set up rules that retrieve data from prior periods. Historical rules can be used in formulas and fictitious calculations.

Historical Usage Calculation Method

In PeopleSoft Inventory Planning, a method that defines the set safety stock or minimum inventory level. The usage is based on the review of historical demand over the number of effective periods. The historical demand quantity is determined by one of four methods; maximum possible usage, Lead Time, estimated daily or period use, and static values calculations.

Hold Grade/Step

Grade/step the employee was in prior to receiving a temporary promotion.

Hold Last Equivalent Increase (LEI)

Date held by an employee for this event prior to receiving a temporary promotion. Necessary in order to establish the WGI due date if returning to original grade/step.

Hold Position Description

The new position description number that is the result of a reclassification action prior to the NOA being processed.

Hold Purchase

A flag that tells the system to keep this participant in the purchase process. The hold flag is maintained at the contribution page.

Hold Within Grade Increase (WGI) Due Date

WGI due date prior to an employee receiving a temporary promotion.

Holding Period

Typically refers to the holding period required for ISO's and Qualified Section 423 Purchase Plans, to receive preferential tax treatment on a disposition of shares. See Disqualifying Disposition.

Hours Counting Service

A service calculation that uses actual or generated hours to determine the service credited to a pension plan participant.

Hours Equivalence Service

A service calculation that uses hours to determine service, but that uses a set number of hours per day, week, or other period worked rather than counting actual hours.

HR Action/Reason Category

A group of related job actions—for example, hire and rehire—treated similarly for pension purposes in PeopleSoft Pension Administration.

HRMS Warehouse

See Warehouses.

I

Ignore Plan

Complex event processing feature of PeopleSoft Benefits Administration that enables the user to designate plan types linked to a particular Event Rules/Event Classification combination as being unaffected by Benefits Administration processing.

Ignore Violations

The ability to report over capacity violations but not to score or repair them during the optimization process.

Imputed Income

Theoretical income that a company pays on behalf of an employee but the individual does not actually receive. This “theoretical income” must be added to the employee’s gross wages. In general, imputed income refers to the value of excess Group Term Life or Dependent Life coverage.

In Punch

Indicates start of a shift.

In the Money Option

When the fair market value of the stock is greater than the grant price of an option.

INAIL code

In Italy, the INAIL code is used to classify jobs according to the level of risk associated with the job and the related risk insurance required by the employer. INAIL codes are defined by the employer.

Incentive Pay Plans

In PeopleSoft Workforce Analytics, pay plans that are formula-driven based on the expected results defined at the beginning of a performance cycle. Incentive plans are designed for the individual worker, or for group levels such as teams, business units, divisions, or company-wide. Incentive plans are used for a variety of reasons; including cost control, alignment of employee and shareholder interests, and increased focus on specific performance indicators.

Incentive Plans

Pay plans that are formula-driven based on the expected results defined at the beginning of a performance cycle. Incentive plans can be designed for the individual worker or at group levels such as teams, business units, divisions or company wide.

Incentive Stock Option (ISO)

For an option to be considered an Incentive Stock Option, it must have the following characteristics:

- The option must be granted pursuant to a plan which includes the aggregate number of shares which may be issued under options and the employees (or class of employees) eligible to receive options, and which is approved by the stockholders of the granting corporation within 12 months before or after the date such plan is adopted;
- The option must be granted within 10 years from the date such plan is adopted, or the date such plan is approved by stockholders, whichever is earlier;
- The option is not exercisable after the expiration of 10 years from the date such option is granted;
- The option price is not less than the fair market value of the stock at the time such option is granted;
- The option is not transferable by such individual otherwise than by will or the laws of descent and distribution, and is exercisable, during his lifetime, only by him, and;
- The optionee, at the time the option is granted, does not own stock possessing more than 10% of the total combined voting power of all classes of stock of the employer corporation or of its parent or subsidiary corporation.

Incomplete Punch

A punch that cannot be processed (i.e. missing employee ID, invalid date or time).

Incremental Budgeting

A budgeting option during budget development that uses prior year actual or budget values as a basis and then applies a percentage that increments the base. PeopleSoft Budgeting-specific.

Incumbent

An employee currently assigned to a position.

Indirect Compensation

Typically involves non-cash types of compensation awarded to the individual in exchange for their contribution to the organization. Common types of indirect pay include health and welfare benefits (for example, medical, dental, vision, long-term disability, short-term disability, unemployment insurance), payment for time not worked (for example, holiday, vacation, sick), and employee services and perquisites (for example, club memberships, parking, holiday gifts).

Indirect Cost

A cost that is assigned by management to an activity or a cost object. An example is the cost of office space assigned to an activity.

Individual Occupational Requirements

Requirements, e.g., experience or education, for particular occupational series of positions within a series and are used in conjunction with a group coverage (generic) standard.

Individual Retirement Record (IRR)

Used by the Office of Personnel Management (OPM) as the basic record for determining the retirement benefits payable to separated federal employees and their survivors. Employees covered by the CSRS retirement plan require SF-2806. Employees covered by the FERS retirement plan require SF-3100. In addition, the SF-2806-1 and SF-3101 are used for corrections to the IRR. See also Correction to IRR.

Inherit Control Group Policies

In PeopleSoft Inventory Planning, a feature that controls whether the policy for an item is set explicitly or defaults from the associated **Policy Control Group**. A series of check boxes enable you to define which policies to inherit.

Initial COBRA Events

The event which makes an individual eligible for COBRA coverage. Typical initial COBRA events include loss of benefits eligibility due to termination, reduction in hours, retirement, and military leave, as well as divorce, death of employee, and Medicare entitlement. See COBRA and Secondary COBRA Events.

INSEE (National Institute for Statistical and Economical Studies) Codes

INSEE is an official statistics and economics organization in France. INSEE codes for your French company's organizations are used in regulatory reporting.

INSEE PCS (Classification par Catégorie Socio-Professionnelle) Code

Each PeopleSoft Human Resources French Jobcode is linked to a four-digit INSEE PCS, or social/professional classification code.

In-Service Date

In PeopleSoft Asset Management, the date upon which an asset is placed in service. In-service date is used in conjunction with an asset's prorate convention to determine Begin Depr Date.

Inservice Placement

Includes a noncompetitive action in which a position is filled with a current or former competitive service employee through promotion, reassignment, change to lower grade, transfer, reinstatement, reemployment, or restorations. Inservice placement also includes noncompetitive conversion f appointees whose Federal excepted positions are brought into the competitive service under Title 5 CFR 316.702, and Department of Defense/Nonappropriated Fund (DOD/NAF) and Coast Guard NAF employees whose positions are brought into the competitive service.

Insider

An officer, director or principal shareholder of a publicly owned company and members of his or her immediate family. This category may also include other employees of the company and people who obtain nonpublic information about the company.

Insider Trading

When a person trades a security while in possession of material non-public information in violation of a duty to withhold the information or refrain from trading. The securities law broadly prohibits fraudulent activities of any kind in connection with the offer, purchase, or sale of securities.

Instance

A row of data on the Positive Input table. Instances of positive input can be entered manually, or can be system generated. They can also be received from other applications, such as PeopleSoft Time and Labor.

Integration Level

The salary level in a defined benefit excess plan at which a higher benefit rate becomes applicable. For example, the following formula uses a $10,000 integration level: 1% of Final Average Earnings up to $10,000 plus 1.75% of Final Average Earnings over $10,000.

Integration Template

A high-level template that defines the integration between PeopleSoft Projects and your other financial applications. Each integration template you create defines a specific set of business units from your other financial applications. Each project is then assigned an integration template containing this preset integration information. You can use Integration Templates to set up joint ventures, and new transactions added to that project will reflect the business units defined in the integration template.

Intensity

The cost for each unit of the activity driver.

Interest

Some companies pay interest on the monies that are being withheld from employees' paychecks. The interest plus the employees' stock purchase contributions are used to purchase stock at the end of the purchase period.

Interest Rate Modeling

An FSI feature that allows you to model interactively interest rate scenarios for Asset Liability Management, and to run rate scenarios and analysis in real time.

Interest Rate Sensitivity Model

In the financial services industry, this support module describes in granular terms how a group of customers holding a specific type of instrument with a particular interest rate will respond to changes in interest rates in the market.

Interface Loader

An SQR delivered with PeopleSoft Asset Management that is used to transfer load lines into the PeopleSoft Asset Management loader tables.

Internal Data

Data from PeopleSoft ERP systems, or other legacy ERP systems used by your organization.

Interpolation

To calculate a value of a function, or series, between two known values.

Interunit Account

The account for each business unit to which other business units in the same corporation refer when they need to distribute amounts across business units. These accounts are used to keep the individual ledgers in balance when a single transaction affects multiple business units.

Inter-Unit Drivers

Drivers that provide a means of establishing relationships between the cost objects of one organization with the supporting activities of the organizations that share business units and models.

Interunit Transaction

A transaction that involves moving amounts from an account in one PeopleSoft General Ledger business unit to an account in another General Ledger business unit.

InterUnit Transfer

A transfer that occurs between different business units.

IntraUnit Transfer

A transfer that occurs within one business unit.

Intrinsic Rewards

A reward that is generated by the worker internally such as job satisfaction, as opposed to Extrinsic Rewards which are tangible rewards.

Inventory Adjustment

A process that enables you to change the quantity of an item in the inventory system to match the actual physical quantity found in the **Storage Location**.

Inventory Business Unit

Usually a warehouse. You establish a separate inventory business unit (IBU) for any one of the following reasons: 1) You want on hand visibility to a specific location of your business that manages inventory. 2) You want to define replenishment rules for a specific location of your business that manages inventory. 3) You maintain standard and average costs in a specific location of your business that manages inventory.

Inventory Cost Element

A cost that can be associated with inventory items and inventory transactions. Examples include freight, overhead, and transportation. Each cost element has a unique cost code.

Inventory Item

A tangible commodity that is stored in an Inventory business unit (Ship From warehouse).

Inventory Location

See **Storage Location**.

Inventory Policy

In PeopleSoft Inventory Planning, a set of rules that controls how inventory policy values are calculated for items. Inventory policy is defined at the **Policy Control Group** and stockkeeping-unit levels. The elements that make up inventory policy are order quantity, safety stock, **Reorder Point**, and minimum and maximum policies.

Inventory Transaction

An event that moves inventory into, within, or out of the inventory business unit. Examples include material transfers, inventory adjustments, and standard issues.

Inventory Transaction Group

An identifier that categorizes transactions by type for costing purposes. For example, you can group all types of interunit transfers together.

Invoice Format Identifier

An identifier for the formatting options that determine the sorting and summarization levels of invoice information.

IRC 423 (Internal Revenue Code 423)

The section of the IRC that defines a Qualified Employee Stock Purchase Plan.

IRR Fiscal Data Accumulation

This report accumulates all retirement deductions for employees, as well as any LWOP and any basic pay that was received when an employee was not covered by the CSRS or FERS retirement plans.

IRR Remarks

Special remarks that are documented on an employee's IRR. IRR Remarks can be set up ahead of time and can be system-entered text or employee-specific.

IRR Status

IRRs can be in pending or final status. Those in pending status can be updated and corrected. A final status indicates that the IRR has been processed and can't be updated or corrected except through a Correction IRR or a Supplemental IRR.

IRR Worksheet

A preliminary IRR form that enables an agency to print a pending IRR for a separated employee, review it and make corrections, if necessary. Agencies can also use the IRR Worksheet to view a current IRR for an active employee.

ISO IRS $100K Limit

The limit the IRS places on the exercisable value of Incentive Stock Options (ISOs) of $100K per calendar year based upon the fair market value at the time of grant (Section 422 of the Internal Revenue code).

ISO to NQ Grace Period

The period of time after which an Incentive Stock Option is treated as a Non-Qualified Stock Option for tax purposes upon the termination of employment according to Internal Revenue Code Sections 421 and 422. Depending on the termination reason the option is treated:

- If the termination reason is for any reason other than death or disability, and an exercise occurs more than three months from the termination date, the system withholds taxes as if the option is a non-qualified stock option.
- If the termination reason is disability, the system withholds taxes if an exercise occurs more than twelve months from the termination date.
- If the termination reason is death, the system always treats the option as an ISO.

Issue

See **Material Issue**.

Issuer

A legal entity that has the power to issue and distribute a security.

Item

See **Inventory Item Planning Item** or **Receivables Item**.

Item Content Provider

Third-party software consisting of web-based catalogs of item and price information. These systems benefit the design and purchasing of new products by accelerating item location, maximizing design reuse, and reducing acquisition costs. PeopleSoft Purchasing, Engineering, and Inventory integrate to Item Content Providers, and the information is used by many other PeopleSoft applications.

Item Rounding Rules

A set of rules determining how fractional values are rounded so that calculations result in whole numbers. Rounding rules are used in conjunction with **Quantity Precision Rules**.

Item Simulation

In PeopleSoft Demand Planning, a process that enables you to interact with the forecast in a manageable manner and perform "what-if" analysis by comparing the effects of different forecast models.

Item Type

An identifier that defines inventory items at a very high level, and may include sets of Item Families. For example, the families Computer Items and Office Furniture might be categorized by types like Outside Manufacturing, Finished Goods, and Work In Progress.

Item-Specific Conversion Factor

A conversion between the same two units of measure when the measurements have a different value for an item. For example, a conversion between packaging unit and stocking unit.

Iterative Processing

Refers to a concept on only re-calculating those payees who have had changes and need to be recalculated (if you choose to run your payroll multiple times before actually finalizing it). This concept saves you a lot of time as you only have to recalculate those payees who have had a data change or who you indicate you would like to be recalculated.

J

Java Server Handlers (JSH)

The JSH manages network connectivity, making service requests from the Jolt Repository, and translating Tuxedo buffer data into the Jolt buffer.

Java Station Listeners (JSL)

The JSL handles the work of the client connection, tracking client messages, and session handoff.

Job Code

An ID for a job as defined on the Job Code table.

Job Code Components

The pay components assigned to a job code by associating rate codes with job codes on the Default Compensation page or the Non-Base Compensation page of the Job Code table.

Job Code Cost

Evaluation of salaries for specific job codes.

Job Compensation Rate

The compensation rate of the corresponding job row.

Job Events

Actions relevant to an employee's employment—such as a hire, transfer, or termination—that can affect benefit program or plan eligibility. Used by PeopleSoft Benefits Administration. See Event Class.

Job Order Cost Accounting

A cost accounting method that attempts to develop a discrete cost for each job performed or product produced. Only the material, labor, and overhead required to complete the job are attributed to the job cost.

Joint and Survivor Payment Option

A form of pension payment in which benefits are paid for the life of the participant and a beneficiary. Should the beneficiary outlive the participant, the benefit continues (often in a reduced amount) for the life of the beneficiary.

Joint Staffing Report

In the United Kingdom governmental agencies are required submit the Joint Staffing Report. Although it is mainly designed for government sector organizations, commercial organizations may also use this SQR to provide a summary of their staffing by department, job code, gender and full/part time employment status.

Jolt

A BEA/Tuxedo companion product that runs on an application server domain and is used to listen for Web Client Java requests and transfer them to Tuxedo.

Journal Code

The second highest level of three categories for defining a financial transaction (or document), necessary when using document sequencing. Examples of journal code are domestic sales and export sales. This category is preceded by journal type and followed by document type.

Journal Generator Template

A table containing defaults to be used in journal generation. PeopleSoft Asset Management and Billing require one journal generator template for each transaction type.

Journal Line

A record storing a double-sided, balanced entry for a given journal. A single journal usually includes multiple lines. The sum of the monetary amounts for the journal lines in one journal totals zero (debits = credits).

Journal Template

A list of the characteristics of the general ledger journal entries that will be created from your PeopleSoft Receivables system.

Journal Type

The highest level of three categories for defining a financial transaction (or document), necessary when using document sequencing. Examples of journal types are sales journal and purchase journal. This category is followed by journal code, then document type within the journal code.

Journal Voucher

A PeopleSoft Payables voucher that enables you to make accounting entry modifications while keeping your PeopleSoft General Ledger and Payables systems in sync. Like the adjustment voucher, the journal voucher is linked to an existing voucher.

K

Kanban ID

A unique identifier used to track Kanban cards and replenishment requests when using PeopleSoft Flow Production.

Keep Ledgers in Sync

An option in PeopleSoft General Ledger that defines how a transaction should be posted—to all ledgers in a ledger group as opposed to only a single specified ledger.

Key

See **ChartKey**.

Key

One or more fields that uniquely identify each row in a table. Some tables contain only one field as the key, while others require a combination.

Key Performance Indicator (KPI)

KPI is used by the PeopleSoft Performance Management analytical applications. KPIs are high-level measurements of how well an organization is doing in achieving critical success factors. A KPI defines the data value or calculation from the Data Warehouse tables upon which an assessment is determined.

KPI (Key Performance Indicator)

See Key Performance Indicator.

Knowledge, Skills, And Abilities (KSA)

Also known as Competencies, these are attributes required to perform a job and are generally demonstrated through qualifying experience, education, or training. *Knowledge* is a body of information applied directly to the performance of a function. *Skill* is an observable

competence to perform a learned psychomotor act. *Ability* is competence to perform an observable behavior or a behavior that results in an observable product.

L

Labor Costs

Actual expenditures associated with *salary* portion of time reporter expense.

Labor Dilution

A process that occurs after the Labor Distribution process in PeopleSoft Time and Labor. The labor dilution process takes the costs that the payroll system has calculated for payable time, determines an average or rate per hour, and applies the average amount evenly across all reported hours for the day.

Labor Distribution

The process of distributing payroll expense to the corresponding payable time entries generated in PeopleSoft Time and Labor.

Labor Distribution Amount

An actual labor cost associated with reported time.

Last Equivalent Increase (LEI)

Reflects the effective date of the last step received in grade or the last promotion, whichever is most current (does not include QSI). Used as the basis to establish an employee's WGI due date.

Last Physical Counting Event

The last date the inventory item was counted. This information is stored with each inventory item.

Last Purchase Date

The item's most recent purchase date in the inventory business unit.

Last Putaway Date

The item's most recent putaway date in the inventory business unit.

Last Putaway Document Number

The item's most recent putaway document identification number in the inventory business unit.

Last Receiving Date

The item's most recent receipt date in the inventory business unit.

Last Shipping Date

The item's most recent ship date in the inventory business unit.

Last Shipping Document Number

The item's most recent shipping document identification number in the inventory business unit.

Law Enforcement Officers (LEOs)

Positions within the Federal government involving law enforcement. Under FEPCA, many of these positions are entitled to additional special pays.

Lead-Time Estimated Usage

An inventory planning method for calculating historical usage of an item. The historical demand is prorated on a daily basis and then multiplied by the number of days lead time for each effective historical period. The maximum period value is then used as the safety stock or minimum stock level. This method should be used for items that have a steady demand pattern throughout each period.

Lead-Time Period Usage

An inventory planning method for calculating historical usage of an item. The purchase lead time is rounded up to a specified number of periods. The historical demand is calculated as the maximum usage during these periods and the safety stock or minimum-stock level is set to this value.

Leave

Time entitled to an employee as a benefit, such as, Sick, Vacation, STD, and LTD. This process is managed by HRMS (see Time Reporting).

Leave Accrual Processing

Processing of leave accruals is used to maintain employee leave balances. All leave benefit plans accrue leave by length of service or number of hours worked. Leave accrual processing is used to determine the employee's leave accrual award and resulting leave balance.

Leave Accruals

Hours that employees earn to use at another time, such as annual leave and sick leave.

Leave Plan

A method for earning and managing leave time.

Leave Without Pay (LWOP) Total (Cumulative)

An employee's cumulative number of hours of leave without pay (LWOP).

Ledger Group

In PeopleSoft General Ledger, a group of ledgers consisting of one primary ledger andsecondary ledgers.

Ledger Mapping

Ledger mapping is a process that enables you to relate expense data from your general ledger accounts to resource objects. Multiple ledger line items can be mapped to one or more resource IDs. You can also use ledger mapping to map dollar amounts (referred to as rates) to business units. You can map the amounts in two different ways: an actual amount that represents actual costs of the accounting period, or a budgeted amount that can be used to calculate the capacity rates as well as budgeted model results. In the PeopleSoft Enterprise Warehouse (EW), Ledger Mapping enables you to map general ledger accounts to the EW Ledger table.

Ledger Template

A table containing records and fields common to all ledgers that ensures that all ledgers specified in a ledger group share the same physical layout.

Ledger Type

The unique combination of a single ledger, scenario, and fiscal year. Multiple ledger types make up a ledger type set.

Ledger Type Set

A collection of ledger types, the members of which will represent the members of your ledger type dimension.

Legend ID

A way of recording information that is displayed upon the Issuance Instruction Report. Can be used to record a notice that should appear on the back of a stock certificate indicating that the shares represented are "Restricted Securities." Can also be used to indicate how shares should be processed, as in the case of Swaps, Trades, Repurchases and SAR Exercises.

Level

A section of a tree that organizes groups of nodes.

Defines a set of **Forecast Item** with a common key structure. Each level is related in a hierarchical definition with other levels in the view. A level definition contains descriptive and control data that relates to the operation of the forecast at each level within the view.

Level

The section of a tree that organizes groups of nodes.

Level Income Payment Option

An annuity form of pension payment in which payments are increased in early years (prior to eligibility for Social Security benefits) and decreased in later years when Social Security benefits are also received. The goal is to provide a relatively constant total retirement income both before and after Social Security eligibility.

Life Profile

In PeopleSoft Demand Planning, a feature that enables you to establish product forecasts based on predefined patterns in an item's life cycle.

Lifecycle (of Reported Time)

A representation of time through the various stages of Time and Labor; includes processing of current, future, and previous period time from scheduling and time capture through Time Administration and distribution.

LIFO (Last In First Out)

Method used by companies to record Disqualifying Disposition Income. If a company uses this method they record the optionees disposition of shares by attributing the shares to the most recent exercise, purchase or release dates for which shares remain available for sale.

Line-Item Budgets

The budget amounts associated with ChartField distributions that make up an organization's budget. Line-item budgets include personnel costs as well as operating and maintenance costs. They also include revenue estimates. PeopleSoft Budgeting-specific.

Line Schedule Editor (LSE)

PeopleSoft Production Planning utility or tool that displays production tasks for multiple products on multiple resources across multiple periods of time.

Literal Mapping

In PeopleSoft Demand Planning, a mapping option for formatting data that is common to all records being imported. This enables you to set an available field value for all the loaded rows.

Load

The feature that initiates a process to automatically load information into a PeopleSoft application—for example, populating the PeopleSoft Benefits database with plan-level election information.

Load Activation

Load Activation enables you to specify exactly which part of your Data Mart to build, including security. You set up load activation on the Load Activation page.

Load Planning

The PeopleSoft Inventory feature that picks, packs, and ships orders by Load ID. Load Planning is also used to estimate shipping weight, volume, and charges.

Loader Table

Any table in PeopleSoft Asset Management used to store load lines before they are loaded into the system as open transactions. The loader tables comprise INTFC_FIN, INTFC_PHY_A, and INTFC_PHY_B.

Loan Exercise

A form of cash exercise, typically requiring a loan agreement and a promissory note.

Local Code

In PeopleSoft Demand Planning, a type of validation used for a user-defined field code. If a user-defined field is marked to require local table validation, **User-Field Code** are used to determine the list of valid values for the field.

Local Functionality

Local functionality is the set of information in PeopleSoft HRMS that is available for a specific country. You can access this information when you click on the appropriate country flag push button in the global window, or when you access it by a local country menu.

Location Accounting

An accounting method that captures and records material movement within the warehouse, providing accounting visibility based on where the inventory resides. You can designate certain **Storage Area** as raw material, WIP, or finished goods by assigning the corresponding account ChartField (account, department, product, and project ID) to the storage area. All inventory locations in a storage area use the storage area account.

Location Code

Locations enable you to indicate the different types of addresses a company has—for example, one to receive bills, another for shipping, a third for postal deliveries, and a separate street address. Each of these addresses has a different location number. Every customer role must have a primary location, which will be used throughout the system on all panels that display a customer address. The primary location—indicated by a *1*—is the address you use most often when contacting the customer, and may be different from the customer's main address.

Location Summary

A Picking Plan option that sorts the picking plan according to the highest-level sort options defined and prints the order lines and the total item quantity to pick from each **Storage Location**. Because the layout of the printed report reflects the actual positions of stock to be picked, personnel can follow a serpentine path through the warehouse, fulfilling all orders on the picking plan without revisiting locations.

Lock for Confirm

A flag on the Pay Line record that enables users to access the database 7 days a week, 24 hours a day, without affecting or interrupting payroll processing. Issues a warning message "A payroll is currently in process for this employee. This data will not be processed until the next payroll."

Log file

One way that you can monitor the build process is to review the log files that the build process automatically generates. Keep in mind that the log file is entirely separate from the script file; do not confuse the two. How much information that the log file contains is up to you. You can set up your logging so that all status (both good and bad) appears in the log, or you can specify that just the errors or warnings appear in the log. This section describes the options you can specify in regards to the Build log file.

Long-Term Variable Compensation

In PeopleSoft Workforce Analytics, a component of direct compensation that consists of long-term payments to an employee in the form of stock programs, and deferred compensation.

Lookup Codes

In the financial services industry, these are user-defined codes that enable the system to define and categorize incoming Instrument table information. They also provide a means for you to report on specific data, such as treasury position, balance type, and ledger account.

Lot Status

The status assigned to a lot. In PeopleSoft Inventory, a lot's status can be Hold, Open, Rejected, or Restricted.

Lump Sum

A tax method that determines withholding based on the Canadian Lump-Sum tax table.

Lump Sum Payment Option

A form of pension payment in which some or all of a participant's benefit is paid as a single sum.

Lump Sum Reporting

A Time and Labor process that enables you to report time in a lump sum of hours or units for a single Time Reporting Code, and quantities of time. The system uses a batch process to gather the information you enter, perform edits, and update the daily time tables. The system uses the default assignments you establish for workgroups, taskgroups, shifts and so on.

M

Maintenance Worksheet

A work space for creating write-offs, matches, or adjustments to clean up posted items.

Manage Base Pay Structure

See Base Pay Structure

Manage Compensation Planning

A PeopleSoft Workforce Rewards module that facilitates modeling and analysis of compensation costs across organization units, specific job classifications, or groups. You can focus on the impact of changes to workforce size, or on changes to fixed and variable compensation elements, and determine their effects on current and future payroll costs.

Manage Market Compensation

A PeopleSoft Workforce Rewards module you use to match your company's jobs to similar jobs found in published market compensation surveys. You then calculate a target market rate based on a weighted average from multiple surveys. This market rate is then used to assess your company's gap to market and to perform cost impact analysis.

Manage Retention Planning

A PeopleSoft Workforce Rewards module that enables organizations to analyze the factors that lead to employee turnover, and how retention of key employees affects business performance and goals.

Manual Checks

Any checks calculated and prepared outside of the PeopleSoft Payroll system that you must enter into the system manually.

Manual Count

A PeopleSoft Inventory procedure in which you enter the actual count data and then create the counting event with its header, item records, and count quantities.

Manual Events

Events that are inserted by the user manually through the BAS Activity table. Events are actions that occur, which potentially change employee benefit coverage eligibility—see Event Class for more information. Used by PeopleSoft Benefits Administration.

Manufacturing Cost Element

A particular category of an item's cost. For example, when you produce a subassembly that has a cost of $100, the cost can be broken down further into material costs, labor costs, and overhead costs.

Manufacturing Execution Systems (MES)

Third-party system that enables detailed planning and execution of production activities from production order release to completing finished goods. PeopleSoft Manufacturing integrates to MES.

Manufacturing Task

Any job that can be performed within your manufacturing facility. A manufacturing task is associated with the work center in which the task is completed.

Map File

A file that defines the relationship between fields in a third-party system and PeopleSoft Demand Planning tables.

Mapper Type

This defines whether you are mapping actual or budgeted general ledger line items to resource ID within PeopleSoft Enterprise Performance Management.

Marginal Tax Rate

The tax rate that applies to the next dollar of income generated.

Market Compensation

A compensation review process in which you match your company’s jobs to similar jobs found in published market compensation surveys, for the purpose of establishing new target market rates. Also referred to as Market Based Pricing or Market Analysis.

Market Capitalization

The value of a corporation as determined by the fair market value of its issued and outstanding common stock. It is calculated by multiplying the number of outstanding shares by the current fair market value of a share. Analysts look at market capitalization in relation to book, or accounting, value for an indication of how investor’s value a company’s future prospects.

Market Rate

Compensation rates, usually for regular base compensation or total cash compensation, found in published salary surveys. You use the Market Compensation module in PeopleSoft Workforce Rewards to age and weight this data, to create market rates you can compare against your organization's current pay rates.

Mark-to-Market (MTM) Model

In the financial services industry, the reevaluation of a portfolio's position at current market levels.

Market Variance

A comparison of the difference between an individual's, or group's, actual compensation, and available market compensation data for a comparable population in industry. Market compensation data is usually tied to job codes, and comparisons are usually made between similar jobs. Although the variance to market can be evaluated for any of the compensation components in the Compensation tree hierarchy (such as Total, Direct, or Base), market compensation data is most typically available for, and used in evaluating Base Pay (Base Salary). The main point of reviewing the market variance is to evaluate how well your workforce is paid in comparison to both prevailing compensation in industry, and your own organization's compensation strategy.

Mass Adjustment

A process of applying an amount or percentage change to one or many line item budgets at once. PeopleSoft Budgeting-specific.

Mass Cancellation of Requisitions and Purchase Orders

A utility that allows you to select and cancel groups of requisitions and purchase orders. You can use this utility during the year as well as at year-end in preparation for closing. The utility enables you to specify ChartField criteria for selecting documents for cancellation. For example, you can select all requisitions or purchase orders for a particular fund and organization, which have a remaining balance. Then you may select a subset of those records to approve for cancellation.

Mass Change

A user-configurable entity that defines the movement of data between the tables that store your business information. Mass Changes enable you to define the criteria by which you move or replace data in your tables. Based on the configuration of your system, Mass Change dynamically builds data access and gives you complete control over your system processing.

Mass Change Template

The foundation for defining mass changes. Mass change templates enable you to control which fields will be available for the operator to specify when defining a mass change, and whether those fields will be used as selection criteria or defaults.

Mass Change Type

The building blocks used in defining mass change templates. Mass change types specify which records the resulting mass change will select From the database, alter, and subsequently write back To the database. They also set up system field defaults that run behind the scenes to ensure that this mass change is processed correctly.

Mass Validate Metadata Utility

A PeopleSoft Enterprise Warehouse utility that enables you to validate, but not compile, Metadata objects. Mass Validate certifies all "as of dates" created for Filters, Constraints and DataSets for the specified run date. This utility helps ensure that your Metadata is valid at run time and increases your chance of a successful engine run.

Match

A process in PeopleSoft Workforce Planning, by which the system compares the roles, competencies, and accomplishments in the current competency inventory, with the requirements of a given competency strategy.

Matched Punches

A period between two consecutive punches during which some activity happens measured intervals.

Match-Funding

In the financial services industry, Match Funding refers to funding an asset with a like (term to maturity) liability. This helps an organization apply the appropriate funds transfer price. Although the actual asset might be funded with shorter-term liabilities, it does provide a better measure of financial performance for that asset, such as Risk Adjusted Return on Capital.

Material Costing

An inventory accounting method that assigns a cost to items in inventory. These costs can be assigned equally across all items or tracked individually for each item.

Material Issue

An event that triggers stock fulfillment requests for items in inventory.

Material News

Company news that could be expected to affect the value of a company's securities or influence investors' decisions. Material news includes information regarding corporate events of an unusual and non-recurring nature, news of tender offers, unusually good or bad earnings reports, and a stock split or stock dividend.

Material Release

A PeopleSoft Manufacturing process that—after material has been picked—decrements on hand inventory balances for the inventory storage areas and increments inventory to the WIP

locations defined by the routing or production area. The process also changes the production ID's or production schedule's status from Released to In Process.

MAX Method

See Maximum Method Policy.

Maximum Compensation Hours

The greatest number of hours to be paid for a specified TRC (see Time Reporting).

Maximum Lead-Time Usage

In PeopleSoft Inventory Planning, a policy control value that sets the safety stock level to the maximum quantity required during the lead time. This method is normally used when the demand for an item is low or intermittent but sufficient stock must always be available.

Maximum Method Policy

In PeopleSoft Inventory Planning, a policy that controls the way in which the system determines a reasonable high limit for the maximum inventory level of an item. The system provides warning messages when the inventory level exceeds the maximum level.

Maximum Taxable Wage Base

An annual earnings threshold used for Social Security purposes. Pension plans sometimes provide different levels of pension benefits for earnings above and below the Maximum Taxable Wage Base.

Measure

A measure represents the amounts brought into a cube—the numerical data.

In data warehousing, a Measure is a field type used interchangeably with fact. Measures are types of amounts. Any numeric field you want to apply a Data Manager rule against should be a measure.

Measure ID

In the Define Market Compensation module of PeopleSoft Workforce Rewards, a Measure ID is the identification code for a measure. For market compensation surveys, the Measure ID describes the percentile for each type of pay, as well as the regression statistic type. In PeopleSoft Workforce Analytics, for Benchmark Surveys, the Measure ID describes the type of benchmark.

Measure Value

In PeopleSoft Workforce Rewards, Measure Value is the calculated market rate value from market compensation surveys for a given percentile of a market rate, and for a given scenario and job code. This is the annual rate you compare against the compensation paid for similar jobs in your company. The Measure value can also be the regression statistic value used for

Regressing Market rates. In PeopleSoft Workforce Analytics, for Benchmark Surveys, the Measure Value is the delivered Benchmark Ratio.

Member

A member is the OLAP equivalent of a node or detail value on a PeopleSoft tree. A member is a single item within a dimension, such as a single product name, department ID, or part number. Member names must be unique, even across dimensions. Cube Manager uses the term Dimension Field Mapping to identify members, dimension parents, and label mappings.

Merchant

In PeopleSoft eStore and Mobile Order Management, a level of online (web or wireless device) display and order processing controls set by the seller. See also Merchant Variant.

Merchant Variant

Associated with a user ID, a subordinate level of merchant controls that enable customers to access different variations of the same PeopleSoft eStore website. In PeopleSoft Mobile Order Management, variations are primarily used to differentiate order processing options and fulfillment methods or locations.

Merit Matrix

In PeopleSoft Workforce Analytics, a matrix used to define the amount rules for base compensation increases for your workers. A Merit Matrix provides the salary increase parameters for each review rating in a rating scale. The salary increase parameters are expressed in terms of a percentage. The percentage increase amounts are usually structured to express the company's pay strategy relative to employee performance, and the employee's degree of range penetration in their salary range.

Merit Matrix Increase

In PeopleSoft Workforce Analytics, an increase to an employee's base pay awarded based upon a Merit Matrix.

Message definition

The object definition specified in Application Designer which contains message information for PeopleSoft's Application Messaging system.

Metadata

Information about data. Metadata is the information a database or application stores to describe your business data. At its simplest, metadata defines the structure of a data field—its data type and size, for example. Metadata can also describe more complex data relationships, such as the rollup structure for a chart of accounts. Reporting and analysis tools should be able to use this metadata to let users access data just as they would from within the application, without having to understand how it is stored.

For Enterprise Performance Management, metadata is used to describe the data stored in the PeopleSoft Enterprise Warehouse. There are different types of metadata, for example, TableMaps, DataMaps, and constraints. You typically define these when you set up the warehouse; however, Metadata (particularly constraints and DataMaps) is used to develop business rules that manageaspects of the dimensional models. Metadata enables technical users to define relationships between warehouse tables and enables business users to easily identify the data that interests them without having to know the database structure.

Meta-SQL

Meta-SQL: Meta-SQL constructs expand into a platform-specific SQL substrings. They are used in functions that pass SQL strings, such as in SQL objects, the SQLExec function, Application Engine programs, and so on.

Metastring

Metastrings are special expressions included in SQL string literals. The metastrings, prefixed with a percent (%) symbol, are included directly in the string literals. They expand at run time into an appropriate substring for the current database platform.

Method

A method can only be executed from an object, using dot notation. You have to instantiate the object first, before you can use the method.

For Enterprise Relationship Management, a method is the algorithm or formula that defines how the budget amount for a line-item budget is calculated or how it is derived if a calculation is not necessary. Types of methods include amount per FTE, itemization, annual percent growth rate based on a historical figure, and number of units multiplied by cost per unit. PeopleSoft Budget Planning-specific.

Method Amount

The amount for a line item budget resulting from the application of a method. It represents the budget amount after the method is applied to a line-item budget but before any adjustments or allocations are applied. PeopleSoft Budgeting-specific.

Method Base

The defined value to which a method is applied, if applicable. Not all methods require a base. PeopleSoft Budgeting-specific.

Method Driver

The factor used in a method's algorithm. For the method, amount per FTE, FTE is considered the driver. PeopleSoft Budgeting-specific.

Method of Payment

In PeopleSoft Grants, designates whether a payment is to be through a cost invoice or a Letter of Credit.

Method Parameter

A defined and derived value within a method, which drives an expense or revenue calculation. For the method, Amount per FTE, the number of FTEs is considered the driver parameter. PeopleSoft Budgeting-specific.

Metric

A metric is a calculation of facts. A metric is usually a number, but can be anything you want to measure.

Metric Object Security

Metric Object Security determines whether an individual can see a metric object in a Data Mart.

Midpoint (Pay Range Midpoint)

In PeopleSoft Workforce Analytics, the middle value in a pay range, halfway between the minimum and the maximum, calculated as (Minimum + Maximum)/2.

Midpoint Progression

In PeopleSoft Workforce Analytics, the percentage difference from one grade midpoint to the next higher-grade midpoint, calculated as (Midpoint2 – Midpoint1)/Midpoint1.

MIN Method

See Minimum Method Policy.

Minimum Benefit

See Grandfathered Benefit.

Minimum Compensation Hours

The lowest number of hours to be paid for a specified TRC (see Time Reporting).

Minimum Method Policy

In PeopleSoft Inventory Planning, a policy that controls the way in which the system determines a reasonable low limit for the minimum inventory level of an item. The system provides warning messages when the inventory level drops below the minimum level.

Missed Punch

A punch that is not entered at the scheduled time (see Time Reporting).

mkvdk

Verity's command-line tool used to index a collection, insert new documents, perform simple maintenance tasks like purge and delete a collection, and control indexing behavior/performance.

MLS

Multilingual support.

Modal transfer

Modal transfers allow you to transfer an operator from one component to another component (the modal component) modally; that is, requiring the operator to OK or Cancel the modal component before returning to the originating component.

Modal transfers give you some control over the order in which the operator fills in pages. They are useful for finite tasks related to the main transaction. They are particularly useful in cases where data in the originating component can be derived from data entered by the operator into the modal component.

Model Equivalency Factors

In PeopleSoft Demand Planning, factors that adjust model errors to allow a fair comparison. During the Model Reset process, the errors associated with each of the models are multiplied by their associated factors. The factored errors are then compared to select the model with minimum errors.

Models

In the PeopleSoft Enterprise Warehouse, Models enable replication of an organization's business processes for analysis of cost flow through customers, departments, and channels.

Model Recalculation

In PeopleSoft Business Planning, users may checkout slices of the entire model for their appropriate role. This requires the entire model to be periodically recalculated to incorporate the users changes for dependencies in other areas of the model.

Morphing

Morphing is a technique of automatically transforming the look and feel of an interface based on the needs of an active object. The Application Designer toolbar and menus dynamically transform based upon the type of object definition that is active.

Mortality Table

A table showing rates of death by age. Mortality tables are part of a pension plan's actuarial assumptions.

Moving Average

In PeopleSoft Demand Planning, a model that averages a selected number of the most recent demand periods and creates a forecast of demand for the next and subsequent periods.

Multibook

A functionality supporting the requirement of a company to carry one set of books in their local currency (functional currency) and another set of books in the currency of their parent company (reporting currency). In PeopleSoft General Ledger, multibook functionality is multiple ledgers having multiple-base currencies defined for a business unit, and the option to post a single transaction to all base currencies (all ledgers) or to only one of those base currencies (ledgers). Also commonly known as dual-book.

Processes in PeopleSoft applications that can create both application entries and general ledgers denominated in more than one currency.

Multicurrency

The ability to process transactions in a currency other than the business unit's base currency.

Multidimensional Analysis

A type of analysis that enables you to look at data from many different dimensions, or attributes. You identify the dimensions of the data, then combine the dimensions in various ways. For example, you might identify five dimensions of your sales data: sales, region, channel, product line, and time. Once you've identified the dimensions, you can "slice and dice" the data based on combinations of these dimensions, such as sales in the Western region for the last quarter.

Multidimensional Database (MDDB)

A database that stores data for multidimensional analysis in a proprietary multidimensional format. Users access MDDBs exclusively for reporting and analysis, never transaction processing, so they are optimized for retrieval speed.

For Enterprise Performance Management, a Multidimensional Database stores data for multidimensional analysis in a proprietary multidimensional format. These databases are used exclusively for reporting and analysis, and never transaction processing, so they are optimized for retrieval speed.

Multiple Jobs

Multiple jobs allow you to hire an employee into more than one concurrent job and have them processed through Payroll, Benefits, and Pension. In order to enable this feature, the Multiple Jobs check box must be selected in the PeopleTools Options page.

Multiple-table dynamic tree

The user drills down through a hierarchy of parent and child records.

Multivariate Forecasting Techniques

In Enterprise Planning and Simulation, this is a forecasting method that uses both the recorded history for the target value and the history and forecasts for other variables (causal factors) to infer, not only a forecast for the target value, but also a functional relationship between the causal factors and the target value.

N

National Association of Securities Dealers, Inc. (NASD)

Self-regulatory organization of the securities industry responsible for the regulation of The NASDAQ Stock Market and the over-the-counter markets. The NASD operates under the authority granted it by the 1938 Maloney Act Amendment to the Securities Exchange Act of 1934.

National ID Number

Different countries track some form of National ID for payroll, identification or benefits purposes. For example, German workers are assigned a Social Insurance Number, UK workers have a National Insurance Code, and US laborers have a Social Security Number. Each of these different types of National IDs has unique formatting requirements associated with them as well.

Nature Of Action (NOA) Code

Indicates the type of personnel action being processed.

Nature Of Action Description

Describes the NOA code.

Nature Of Action Effective Date

The date the personnel action is effective.

Negative Amortization

Occurs when a loan payment does not cover the interest due on the loan payment, resulting in an increase of the principal amount.

Net-To-Zero Adjustment

A prior period adjustment where no compensation affecting fields on the pre-existing (original) record are changed by the adjustment.

New Hire Report

In the United States the Personal Responsibility and Work Opportunity Act of 1996 (the so-called Deadbeat Dads law) requires employers to report new hires to specified agencies within a pre-determined number of days from the hire date.

Next Level Item

In PeopleSoft Demand Planning, the **Forecast Item** at the next level that contains the current item as a child. This is the key of the group item at the next level up and is always within the same view.

Next Year

PeopleSoft Benefits term referring to the next open enrollment processing year.

NIC (Numéro Interne de Classement) Code

In France NIC numbers identify the entities inside the same enterprise, and represent an Internal Filing Number.

No Control

A target control that allows the user to submit a budget even if it is not within the planning target and the tolerance levels. The system tracks the budget against the defined planning targets but does not generate any warnings or validations. Users can still compare their planning targets against their budget amounts on the Planning Targets page in Line Item Budgeting.

Node

An individual item on a tree. Nodes summarize detail values or other nodes, and may or may not roll up into other nodes or levels.

Node

A node is a name that you can use to refer to some source of HTML content. In more practical terms, a node is a URI string that defines the database and server to be used when the portal servlet attempts to retrieve content, proxy addresses, and assemble pages.

Non-Base Pay

A pay component not included in the job comp rate calculation. It is used by payroll only in the paysheet calculation. For example, non-base pay can be set up for additional work, holiday pay, bonuses, and so on.

Non-Benchmark Jobs

See Benchmark Jobs.

Noncompetitive Action

An appointment or placement in a position in the competitive service that is not made by selection from an open competitive examination, and that is usually based on current or prior Federal service. A noncompetitive action includes:

- All of the types of actions described under inservice placement, above
- Appointments of non-Federal employees whose public or private enterprise positions brought into the competitive service under Title 5 CFR 316.701; and
- Appointments and conversions to career and career-conditional employment made under special authorities covered in 5 CFR 315, Subpart F.

Nondiscrimination Tests (NDT Tests)

Tests used to help employers ensure that their organization's 401(k), 401(m), and Section 129 dependent care reimbursement plans do not discriminate in favor of highly compensated employees. See Highly Compensated Employees.

Non-Employee

Those workforce resources hired to perform a specific job and/or hired for a specific period of time. Although non-employee time will be entered into Time and Labor for the purposes of managing their Task time, non-employee earnings will not be updated to Payroll and they will not be paid through the Payroll system.

Non-HR Employee [Time and Labor]

An individual employed by the corporation who is administered outside of the PeopleSoft Human Resources system.

Non-Job Event

Actions which result in changes to an employee's personal or demographic information that also affect benefit program and plan eligibility—such as an a state or postal code change, a family status change like a divorce, or a birthdate change. Used by PeopleSoft Benefits Administration. See Event Class.

Non-Productive Time

Any employee scheduled work time spent on tasks (or non-tasks) other than those which the employee was hired to perform. This could include time spent in training, time spent in meetings, travel time, and time spent reporting time.

Non-Qualified Dependent

Dependents such as domestic partners, their children, and other people who do not meet the definition of qualified dependents as presented in IRS Section 152. PeopleSoft Benefits applications enable the creation of benefit programs that offer health and life coverage to non-qualified dependents.

Nonqualified Plan

A plan that doesn't conform to ERISA rules. Employers cannot take a tax deduction for contributions to a nonqualified plan; instead, plan benefits are generally paid directly from the employer's assets.

Nonqualified Stock Option (NQ)

Any option that does not satisfy the conditions of a statutory stock option under the Internal Revenue Code and therefore does not qualify for preferential tax treatment. Generally, companies can design nonqualified options in almost any way they like. Features are:

- The grant price may be less than fair market value (with some exceptions under state law).
- Grants are not limited to employee of the company or subsidiary.
- No taxable income is recognized at the time of grant.
- Options can be granted to anyone (Employees, Consultants and Board of Directors).
- Difference between the fair market value on the date of exercise and the grant price is treated as compensation income.
- In the U.S., withholding tax obligation arises at the time of exercise.
- Company receives a tax deduction equal to the compensation income recognized.

Nontaxable Benefits

Any employer contributions that are not subject to Federal Withholding Tax, such as an employer's portion of a 401(k) plan.

Normal Form of Payment

The payment form associated with the amount calculated by the benefit formula. Pension Administration uses it as a basis for converting to optional forms of payment.

Normal Hours

The hours an employee is normally expected to be at work for any given workweek.

Normal Line Of Promotion (Career Ladder)

The pattern of upward movement from one grade to another for a position or group of positions in an organization.

Normal Retirement Date (NRD)

The date on which an employee is eligible to retire and begin receiving pension benefits. Eligibility for normal retirement is typically based on age only.

Normalized database

A normalized table adheres to certain standards designed to improve the productivity of the database user. Normalization makes the database much more flexible, allowing data to be combined in many different ways.

The standards for a normalized database are called forms, such as first normal form, second normal form, and so on.

Normalized Loss

In the financial services industry, Normalized Loss is the expected loss on a loan and is netted out of the profit and loss statement for management accounting or profitability measurement purposes. Similar to the bank's loan loss reserve, it enables the institution to analyze and account for expected losses on a more detailed level, by financial product.

Northern Ireland Report

In the United Kingdom the Fair Employment (Northern Ireland) Act of 1989 requires private sector employers with more than 10 employees to submit the Northern Ireland report to the Fair Employment Commission annually. The report indicates the religious composition (referred to as Community Background—Catholic, Protestant, Other) of the workforce, job applicants and appointees.

Not To Exceed (NTE) Date

Types are as follows:

- Appointment NTE Date: Indicates the length of time a person may serve in a position.
- Classification Temporary NTE Date: Established temporary date that is used for a temporary classification of a unique position.
- Hospitalization coverage.
- LWOP NTE Date: NTE date is the last day the employee is in leave without pay status. The employee is scheduled to return to duty the next workday.
- Position NTE Date: Indicates the length of time a position is available for use.
- Promotion NTE Date: Specific NTE Date: Specific time for an increase in grade on a temporary basis.
- Suspension NTE Date: Specific time an employee is to be on suspension. No salary is paid for the period.

nPlosion

A PS/ nVision feature that enables you to expand rows and columns in your spreadsheet to underlying details, as in drilldown.

Numeric constant

Numeric constants are any decimal number used in PeopleCode.

O

Object-Based Modeling

Object-Based modeling technology enables you to create parent and child models. In the PeopleSoft Enterprise Warehouse, you set up such models using the Scenario Manager.

Object reference

An object reference is one that uses the current object. For example, in the case of a component, pages within the component are related objects. The menus that use the component are its object references.

Occupant Of Position/Vice

Indicates new position or former occupant of a position.

Occupational Series Code

Designates a grouping of positions similar in work and qualification requirements. They are designated by a title and four digit number (e.g., the Accounting Series, GS-0510).

Off Date

A specific date that is defined as an off day (see Scheduling).

Off Day

A 24-hour period rounded by daybreaker with no associated shifts (see Scheduling).

Off Day Type

A classification of off days (i.e. holiday, plant shutdown) (see Scheduling).

Off-Cycle Processing

The process of calculating and creating a paycheck for one or more employees aside from the normally-scheduled *(on-cycle*) payroll run for their pay group. You typically use off-cycle payroll processing for employees who are being terminated, new hires who weren't entered into the system in time for the last on-cycle payroll run, and employees who received an incorrect paycheck during a normal on-cycle payroll.

Offer Period

This is the period of time in which an employee’s ESPP share price is determined.

Officer

An insider who sits on the Board of Directors and who is also an employee of the corporation. Examples include CEO, CIO, CTO, CFO, COO, Corporate Secretary, and Treasurer.

Official Forwarding Address

An employee's mailing address following separation.

Official Languages Act (OLA)

Canadian federal institutions are required to report on the official languages used in their departments, in accordance with the Official Languages Act (OLA).

Official Personnel Folder (OPF)

The repository of a Federal employee's official documents related to Personnel history.

Official Personnel Folder (OPF) Address

Indicates the address where the Official Personnel Folder is maintained.

Off-Invoice Discount (OI)

A per unit discount deducted from the customer invoice and given by a manufacturer for promotional activities. Off-invoice discounts can originate from a National Allowance or Customer Promotion, and are passed to PeopleSoft Order Management so the discounts are applied correctly during order entry.

Offset Plan

A pension plan where the benefit formula includes an offset of a portion of the participant's Social Security benefits.

OLAP

Online Analytical Processing. OLAP is the multidimensional analysis of application data, performed interactively. The acronym contrasts with OLTP (Online Transaction Processing), which is what most production business application systems do.

OLTP (Online Transaction Processing)

OLTP refers to the applications that perform the business transactions that keep your company running, such as processing invoices or enrolling employees in benefits programs.

Ontario Employment Equity Commission (OEEC)

The OEEC requires employers in Ontario to complete workforce surveys.

Open Enrollment

The scheduled annual re-enrollment of plan participants into appropriate benefit programs and, within those programs, benefit options.

Open Price

The price at which a security starts a trading day.

Open Season

A time period during which Federal employees are open to re-enroll in a specific benefit plan and option. Open Seasons can be scheduled at varying times throughout the year and multiple Open Seasons can occur concurrently with each other. For FEHB processing, it is generally the time period from mid-November through mid-December. For Thrift Savings Plan (TSP) processing, these are semi-annual and are generally held from May 15 - July 31 and November 15 - January 31. Open seasons for FEGLI are infrequent and special notification from the OPM would be issued to all Federal employees should they occur.

Open Transaction

A transaction that has not yet been processed in PeopleSoft Asset Management.

Operation

In PeopleSoft Manufacturing, a job or task performed in a specified amount of time, done in one work center, and using one or more resources.

Operational Data Store (ODS)

A staging area in PeopleSoft Enterprise Warehouse for source application data and pre-processed data for tables optimized for reporting.

OPF Code

Indicates where the OPF is maintained.

OPM

Office of Personnel Management.

Optimization

In PeopleSoft Demand Planning, the process of evaluating and improving forecast model parameters.

Optimize

The process of creating a new PeopleSoft Planning schedule by repairing the violated constraints in a schedule automatically. The Optimizer can be prioritized for meeting due dates, for minimizing overtime costs, and so on.

Option

A contractual right that gives the individual the option to purchase a specified number of shares of stock through an Equity Compensation Plan. Also known as a grant. Regulatory agencies also refer to an option as the right to purchase stock in an employee stock purchase plan. These options are considered granted on the offering begin date.

Option Adjusted Cost (OAC)

In the financial services industry, the difference in the average expected return between an instrument without embedded options that are otherwise identical to the fully loaded instrument and the instrument fully loaded with embedded options.

Option Adjusted Spread (OAS)

In the financial services industry, the average return expected for an instrument, over the short-term risk-free rate, for all projected interest rate paths generated using Monte Carlo simulation.

Option Types

Types of stock options. PeopleSoft Stock Administration supports the following stock option types. Incentive Stock Options (ISO), Nonqualified Stock Options (NQ), Tandem Incentive Stock Options/Stock Appreciation Right (ISO/SAR), Tandem Nonqualified Stock Option/Stock Appreciation Right (NQ/SAR), Restricted Stock Award (RSA).

Optional Forms of Payment

Any alternative forms of payment available to a participant retiring under a pension plan. These can include: annuity options paid over the participant's (and possibly a beneficiary's) lifetime; certain term options paid over a specified number of years; and lump sum options paid out in a single payment.

Options Outstanding

The total number of option shares held by optionees. It is the number of Grants less the number of Exercises, Cancellations, and Expirations.

Order Group

Order groups link order terms that default into sales orders and quotes when you select an order group code.

Order Line Number

The line associated with an order identification number. The order line identifies an item and the requested quantity.

Order Quantity Policy

In PeopleSoft Inventory Planning, a policy that determines how replenishment order quantities are calculated for an item. For example, you can use a static number, provide upper and lower limits, or use an economic order quantity calculated by the system.

Ordinary Income Tax

An individual's tax on earnings from wages, tips, and all other sources except capital gains. Includes option profits upon exercise of non-statutory options.

Origin ID

A code that identifies the location of a payment deposit in PeopleSoft Billing. Origin ID also distinguishes the method of the payment's entry: online, external, or lock box interface. Billing origin ID identifies the remit to origin for billing. This function is mainly used for specifying where the customer should send payment.

Original Option

A stock option that is eligible for repricing. This option has a grant price greater than the current FMV.

OSHA 200 Occupational Injury and Illness Recordkeeping Log

In the United States this record-keeping logbook meets reporting requirements for reporting occupational injuries and illnesses to the Occupational Safety and Health Administration (OSHA). It lists the case numbers and details of each injury and illness that occurred during a calendar year.

Out Punch

Indicates the end of a shift.

Out-of-the-Money

A term used to describe an employee stock option when the current market price is below the option grant price. When an option is out-of-the-money, it costs more to exercise than the underlying stock is worth. Such options are also described as being "underwater."

Output Result Tables

Refer to the database tables that are populated with information at the end of each pay calculation.

Output VAT

VAT collected on sales or outputs.

Outside Scope of VAT

A transaction determined as not subject to VAT. No VAT code is associated with this type of transaction. The transaction is still logged in the VAT transaction table, but no tax is applied.

Outside The Register Appointment

An appointment in the competitive service made under an agency's applicant supply system because either there is not a sufficient number of eligibles on the appropriate register or no competitor inventory exists. Agencies are also authorized to make temporary limited appointments outside the register at grades GS-12 and below.

Outstanding Option

A stock option that still has unexercised (vested or unvested) uncancelled or unexpired shares. Options with a "pending" status are not included. Only options with a status of 'active' or 'suspended' are considered outstanding.

Overlapping Promotions

Multiple customer promotions related to the same customer, and the same product, at the same time.

Override Rate

Cost per hour or unit reported with time used to replace the time reporter's default rate. (see Time Reporting)

Override text

Text not derived from field descriptions.

P

Package level

The top level of organization is the package level. The package is the entire transaction set file, addressed to your company much as a mail package would be.

Page

A page defined in Application Designer as part of a PeopleSoft Internet Architecture application.

Page Assembly

Page assembly is one of the functions of the portal servlet. Page assembly involves intercepting the user's content request, retrieving the content, and properly formatting it using a pre-defined portal template. To complete the page assembly process, the portal servlet merges content from any HTML documents that it retrieves along with the defined template

HTML. The assembled page is then sent back to the user's web browser as a single HTML document.

Page buffer

Consists of rows of buffer fields that hold data for the various records associated with page controls, including the primary records, related display records, derived/work records, and translate table records.

Pagelet

A page designed to appear on a customized homepage. A pagelet is smaller than the typical page dimensions in many PeopleSoft applications. It can be based on either a page designed in Application Designer or on an iScript.

Paired Punches

Two punches for the same employee in chronological order that exists for the purpose of determining the duration between the punches.

Par Value

The nominal or face value of a security. It establishes a price floor below which shares may not be issued. With common stock, the company issuing the stock sets par value. Par value has no relation to fair market value. Some companies issue no par value stock.

Parallel Processing

In the PeopleSoft Enterprise Warehouse, parallel processing is a system function that "locks in" the information you use for processing. This enables the system to run identical or similar processes at the same time without impacting your results. Running concurrent processes greatly reduces the amount of time it takes to run within the system.

Parent Budget

In commitment control, you can build a hierarchy between different budgets, such as summary and detail budgets. Specifying a relationship of parent and child between a summary and a detail budget for purposes of budget inquiries enables you to retrieve information about either budget through the other.

Parent/Child Models

Object based modeling technology enables you to create parent and child models. In the PeopleSoft Enterprise Warehouse, you set up such models using the Scenario Manager.

Parent node

A tree node linked to lower-level nodes or details that roll up into it. A node can be a parent and a child at the same time, depending on its location within the tree.

Parent Task

A higher-level Planning task in a schedule's hierarchy that drills down into subtasks. Its start time is the start time of its earliest subtask, and its end time is the end time of its latest subtask. Planning tasks are distinct from Manufacturing tasks.

Partial Pay

The pay processed whenever a job record has an effective date in the middle of a pay period. Typically, this happens whenever you hire, terminate, transfer, or change the rate of pay for an employee mid-period.

Participants

Individuals who elect to participate in the stock purchase plan.

Participation

The PeopleSoft Pension Administration function that determines whether an eligible employee has met the plan's rules for joining the plan. Generally, these rules are based on age and service criteria.

Participation ID

In the financial services industry, this is a lookup code used by the financial analytic applications to identify the participants (syndicators) involved in, or responsible for, a financial instrument or group of instruments.

Passive Control

A target control allowing the user to submit a budget even if it is not within the planning target and tolerance rules. The system responds by sending an email to the user of the next budget center level indicating that the budget exceeded planning target tolerance levels.

Passive Events

Events that are initiated by a change that has taken place over time, rather than by a direct data entry action. Events are actions that potentially change benefit coverage eligibility. Examples of passive events include an employee’s reaching the age of retirement. See Event Class for more information.

Pattern Reporting

A Time and Labor process that enables you to report a start and stop date, a pattern of one or several time reporting codes, associated hours, amounts, or units and task information once for an employee. The system transforms the information into instances of daily time for each scheduled employee work day based on the employee’s schedule.

Pay

Types of "pay" are as follows:

- Basic Pay: generally, the total amount of pay received during any one calendar year at the rate fixed by law or administrative action for the position held by the employee or judicial official prior to any deductions and not including any special payments or premium pay.
- Gross Pay: total compensation earned by an employee, annuitant, or survivor of a judicial official prior to any deductions. Includes basic pay plus locality pay; availability pay (if any) for LEOs; special payments (if any); an annuity (if any); plus awards (if any).
- Premium Pay: pay provided to an employee as a regular addition to basic pay (e.g., administratively uncontrollable overtime (AUO), availability pay, overtime, night differential, holiday pay, etc.).

Pay Basis

A code indicating the principal condition in terms of time, procedures or criteria, that serves as a basis for computing an employee's pay.

Pay Calculation

Formula that calculates an employee's gross to net.

Pay Calendar

Payroll processing cycle for a given pay group.

Pay Components.

Rows in the compensation record. They build the compensation packages in the compensation record.

Pay Confirmation

Process in which the system updates all to-date totals on the database for earnings, deductions, and taxes for pay groups assigned to a given Pay Run ID.

Pay Entity

A pay entity is the organization responsible for making payments to payees. You can also use a pay entity to define the type of currency to be used when processing calculations. The pay entity is a legal definition of an organization from a payroll perspective. In many cases, an organization and a pay entity are the same. However, PeopleSoft Global Payroll does not define a relationship between an organization and a pay entity.

Pay Frequency

Defines how often employees in a pay group are paid—weekly, biweekly, monthly, and so on.

Pay Group

A set of employees grouped together for payroll processing. It's a way of "bundling" payees for more efficient processing. A pay group is made of payees that the system processes at the same time during a pay run.

Pay Period

The established time segments for which employees in a pay group are paid. Pay Periods are defined by their beginning and ending dates.

Pay Plan

A code that denotes the pay schedule under which an employee is paid , e.g., JS, UG, UJ, etc.

Pay Slip

Either an actual check or an advice notice of a direct deposit. You build these to match your organization's needs. A pay slip is the details of a payment you've made.

Pay Structure

In PeopleSoft Workforce Analytics, Pay Structure consists of a series of pay ranges or grades, each with a minimum and maximum. You develop pay structures to support and reinforce your company's pay strategy (for example, to target the market 50th percentile).

Payable Date

The date that a corporate distribution, such as a dividend, is payable to the record holders of a corporation's securities.

Payable Time

Time that is ready to be collected by the payroll system (see Time Reporting).

PayCycle

A set of rules that define to PeopleSoft Payables the criteria by which it should select scheduled payments for payment creation.

Payee

Any payroll recipient. A payee can be an employee or a non-employee of an organization.

Payee Process Stat Record

A record created for each payee during the payroll process. The system creates one Process Stat record per payee for each calendar.

Payee Section

Type of section that can be added to a process list. A payee section defines a set of elements that is to be resolved for a particular payee.

Payline

Record containing standard payroll information for an employee, such as the amount of regular pay, number of regular hours, additional pay (if any), and tax information and job data.

Payment Interface

An Application Engine process that loads payment information from the Banks Statement tables and the Staging tables to the Application tables and performs various checking and default operations.

Payment Predictor

PeopleSoft Receivables' automatic cash application feature that pairs open items with unapplied payments based on predefined algorithms.

Payment Schedule

A schedule of payment dates for leased assets.

Payment Selection

A process by which PeopleSoft Payables selects scheduled payments that are eligible to be paid in a pay cycle.

Payment Worksheet

The work space in which open items are paired with unapplied payments.

Payroll Certifying Officer

The individual with the delegated authority for approving all items relating to payroll for those employees under his/her authority.

Payroll Process Tables.

Records holding data necessary to process a payroll, such as employee, company, and tax information.

Paysheets

Repository for the raw data necessary to calculate pay for employees, including earnings, hours, deductions, taxes, and accounting data.

PBGC Rates and PBGC Grading

The interest rates published monthly by the Pension Benefit Guaranty Corporation. There is an "immediate" rate that applies once benefits commence as well as a series of "graded" rates—calculated based on the immediate rate—that are used during the time between benefit determination and a deferred benefit commencement.

Pegged Chain

A method the PeopleSoft Enterprise Planning and Production Planning solvers use to determine feasible plans. The method ties tasks together in order to explicitly record which supplies are being used to satisfy which demands. Through this process, the Planning engine first determines which independent demand has the highest priority. Then, it determines the lateness preference ranking for dependent demand tasks.

Penalty

A user-assigned value for constraints that can be violated, determining how the schedule's score will be calculated. Setting the penalty configures the constraint to your priority. Use the Control Page to assign a higher penalty to violations that are more critical to your schedule or a lower penalty to constraints that you can deal with externally.

Pending Exception

Any known exception to an employee's scheduled workday. Pending Exceptions are future dated (future is defined to be for a date under report beyond the last date of the employee's current pay period).

Pending Item

Information in PeopleSoft Receivables that has been entered in or created by the system, but hasn't yet been posted. During the Receivable Update process, the system uses the pending items to update customer balances—either by creating new items or by adding item activity lines to existing items.

Pending Time

Time that has been reported or is assumed to have been reported (based on employee work schedule and calendar date) that has not been used by the business entity. Pending Time may be for past, current, and future pay periods. It is the label for those time transactions that are waiting to be used by the business (for example, approved and unapproved time not yet updated to Paysheets).

Pension Status

An employee's standing with regard to a particular pension plan. For example, employees can be active participants, terminated deferred vested, or in pay status.

PeopleCode

PeopleSoft's proprietary language; it is executed by the PeopleSoft Application Processor. PeopleCode generates results based upon specific actions, based upon existing data or the

actions of a user. Business Interlink Objects are executed by calling the execute() method from PeopleCode. This makes external services available to all PeopleSoft applications wherever PeopleCode can be executed.

PeopleCode Event

An action that an end-user takes upon an object, usually a Record Field, that is referenced within a PeopleSoft page.

PeopleSoft Activity-Based Management (ABM)

A PeopleSoft Analytic Application that aligns organizational costs with operational activities, enabling a coordinated approach to expense and PeopleSoft Activity-Based Management. PeopleSoft Activity-Based Management identifies and assigns operational activities to products, customers, or services.

PeopleSoft Analytic Applications

These are applications within Enterprise Performance Management (EPM) that help you enrich the data in the PeopleSoft Enterprise Warehouse and perform forward looking simulations and scenarios. These applications include: PeopleSoft Activity-Based Management (ABM), PeopleSoft Asset Liability Management (ALM), PeopleSoft Balanced Scorecard (BSC), PeopleSoft Funds Transfer Pricing (FTP), PeopleSoft Risk Weighted Capital (RWC), and PeopleSoft Workforce Rewards. PeopleSoft Funds Transfer Pricing and PeopleSoft Risk Weighted Capital are applications that target the financial services industry (FSI).

PeopleSoft Asset Liability Management (ALM)

PeopleSoft Asset Liability Management provides financial service institutions with the analytical tools to define, measure, monitor and manage interest rate risk, liquidity risk, options risk, and to some extent exchange rate risk. The primary audience for PeopleSoft Asset Liability Management is the financial institution's Asset/Liability Committee (ALCO).

PeopleSoft Balanced Scorecard (BSC)

PeopleSoft Balanced Scorecard converts an organization's vision and strategy into a comprehensive set of performance and action measures that provide the basis for a strategic management system.

PeopleSoft Budgeting

A budgeting application that is a combination of Education and Government (E&G) Budget Planning and Budgets (commercial). This application resides on the EPM database, primarily using the ODS layer of the PeopleSoft Enterprise Warehouse for its data.

PeopleSoft Business Analysis Modeler (BAM)

A multi-dimensional modeling tool used to support several analytic applications.

PeopleSoft Business Planning

A planning application that enables financial executives to model various alternatives and set corporate financial targets to achieve their strategic goals. PeopleSoft Business Planning integrates with PeopleSoft Analytic Forecasting, PeopleSoft Activity Based Management, PeopleSoft Workforce Analytics, and PeopleSoft Budgeting applications.

PeopleSoft Customer Behavior Modeling

A PeopleSoft application that enables you to: a) create a customer profile by extracting customer data from the Enterprise Warehouse; b) create segments and samples from the profile to efficiently target marketing campaigns and further analyze customer behavior; c) append to it additional data from external sources such as demographic, credit or psychographic information; d) use a data mining tool to create a predictive model; e) score the customers in your profile using the predictive model. You can then publish the results to another transactional application.

PeopleSoft Customer Scorecard

A product that provides a pre-defined set of customer-oriented key performance indicators (KPIs), to help you build a scorecard specific to your organization. This scorecard facilitates the measurement and communication of customer satisfaction, customer activity, and objectives across your organization.

See also PeopleSoft Balanced Scorecard

PeopleSoft Enterprise Performance Management (EPM)

Enterprise Performance Management is a comprehensive, integrated analytic business solution designed to increase the value of organizations by enabling people to make better decisions. The PeopleSoft Enterprise Performance Management product line consists of the PeopleSoft Enterprise Warehouse and optional analytic applications and Data Mart products.

PeopleSoft Enterprise Warehouse (EW)

PeopleSoft's data warehousing solution. The PeopleSoft Enterprise Warehouse provides the tools necessary to query, analyze, and present information to provide the optimal environment for business intelligence. It is the central repository for data that will be used with the analytic applications in the Enterprise Performance Management product line, and can also serve as a standalone data warehouse. The PeopleSoft Enterprise Warehouse consists of dimension, fact, reference, and error tables, reporting and ETL tools (Informatica PowerMart), as well as the Operational Data Store (ODS). The tables in the PeopleSoft Enterprise Warehouse are maintained separately from your transaction-based systems to allow for comprehensive analysis of data originating from any Online Transaction Processing (OLTP) or legacy system.

PeopleSoft Funds Transfer Pricing (FTP)

A PeopleSoft Analytic Application that enables an institution to accurately measure and tune profitability. PeopleSoft Funds Transfer Pricing is an interest rate that represents the value of an asset or liability to the institution. PeopleSoft Funds Transfer Pricing is based on market rates, adjusted for risk and cost variables, specific to the institution. By assigning PeopleSoft Funds Transfer Pricing to each item on the balance sheet, the institution can remove the

effects of interest rate volatility from business units, so that profitability measurements are based on factors within their control, that is, credit quality, pricing and product strategy.

PeopleSoft Operations Data Stores (PODS)

See Data Warehouse Tables

PeopleSoft Marketing Insight

A tool that helps you analyze your marketing campaigns and activities. It helps you determine the effectiveness of marketing events based on factors such as number of leads generated, profiles of respondents, campaign return on investment, and campaign forecasted costs to complete.

PeopleSoft Risk Weighted Capital (RWC)

A PeopleSoft Analytic Application that enables the financial services industry to accurately measure capital that has accounted for risk. RWC allocates capital to various levels within a financial institution according to risk, providing the opportunity to measure performance based on how well each business unit, product, customer, or transaction generates income given its perceived level of risk as quantified by the allocation of capital.

PeopleSoft Sales Activity Insight

A tool that helps you analyze key components of the sales process, such as pipeline status, discount analysis, and sales process.

PeopleSoft Support Insight

A tool that helps you determine the effectiveness of your customer service organization. It helps you answer questions such as: Are we effectively handling customer issues? Has contact center performance changed from last year to this year? Which product quality issues are most prevalent?

PeopleSoft Workforce Analytics (WFA)

PeopleSoft's complete workforce analysis solution, which includes the PeopleSoft Workforce Rewards analytical application and the PeopleSoft Workforce Analysis Insight. The complete solution set includes the PeopleSoft Enterprise Warehouse and PeopleSoft Balanced Scorecard products. It helps to manage strategic employee compensation, goals, and competencies, as well as retention.

PeopleSoft Workforce Rewards

PeopleSoft Workforce Rewards is an analytical application you use to align your workforce compensation and retention initiatives with the strategic objectives of your organization. Modules include Market Compensation, Base Pay Structure, Compensation Planning, Workforce Simulation, and Retention Management. You integrate data from multiple internal and external sources, enrich it using rules you define based on any data in your PeopleSoft Enterprise Warehouse, and simulate multiple scenarios of future workforce compensation and

retention activity. You then analyze and evaluate your scenarios, and make actionable decisions you can communicate back to your PeopleSoft ERP systems for execution.

PeopleSoft Portal

The portal bundled with every PeopleSoft 8 application. It provides a simple navigation system, based on existing menu definitions that have been imported into the portal registry. Navigation to content outside of PeopleSoft applications is not provided.

Percent Cycles Without a Shortage

In PeopleSoft Inventory Planning, a method used with safety stock policies. The value is derived from the percentage of replenishment cycles that will complete without a stockout.

Percent Demand Fill

In PeopleSoft Inventory Planning, a method that can be used with safety stock policies. This method defines the percentage of the total quantity ordered that must be filled without a backorder.

Percentage Tolerance

The acceptable percentage difference between expected cycle count quantities and actual quantities counted in PeopleSoft Inventory. This value provides a margin of error for an item during cycle count reconciliation count quantities.

Performance Appraisal Code

Indicates the level of performance of an employee.

Performance Appraisal Due Date

Date established based on the WGI or LEI for the yearly appraisal of an employee.

Period Closing Offset

In Time and Labor, the closing date beyond which this pay period is not considered current any longer, if the period's closing date is different from its end date. You can enter a positive or negative number of days.

Period Method

In PeopleSoft Inventory Planning, a method used to determine how a single static policy value is to be calculated from time-phased results with static policies.

Period of Interest

The maximum period of time containing the data needed to run all the rules in a rule program (see Batch Processing)

Period Segmentation

When an element (like compensation rate) changes mid-period, requiring all other elements in the process list to be calculated multiple times on either side of the date on which the change took place, period segmentation is used. The system calculates each element more than once, using the components that were effective during the different time slices. The system keeps the results of these calculations separate with the object of creating two gross-to-net result sets.

Periodic Processing

In PeopleSoft Pension Administration, any of several batch processes that a plan administrator must run on a regular basis—for example, consolidation of payroll data.

Personal List

A user-created list of products defined in PeopleSoft eStore, used to quickly populate the shopping cart when creating a new order in either PeopleSoft eStore or Mobile Order Management.

Personal Register (Registre Unique du Personnel)

In France, companies are required to be able to produce, at any given time, a Personal Register. For a given establishment, this report lists current employees and employees who left up to 5 years ago.

Personnel Action

Personnel actions are changes to employee data or status resulting from such activities as promotions, transfers, terminations, salary increases, and leaves of absence.

Personnel Representatives (Délégués du personnel)

In France it is mandatory for companies with more than 11 employees to elect personnel representatives who will represent all of the employees before management.

Perspective

In PeopleSoft Balanced Scorecard, a category for organizing critical success factors and key performance indicators. Usually there are four: financial, customer, internal process, learning and growth.

PF Ledger

The PF Ledger (PF_LEDGER_F00) is an important fact table within the PeopleSoft Enterprise Warehouse. The primary function of the PF Ledger table is to support PeopleSoft Enterprise Performance Management reporting. The data that gets posted to the PF Ledger must be accurate and clean.

PF Business Unit

PF Business Units differ from other PeopleSoft Business Units in that they represent functional or strategic areas of an organization, rather than separate legal entities.

Phase Type

A label for the different phases you want to define for a project. Costs can then be calculated by project phases. Examples include planning, startup, construction, and cleanup.

Physical Accounting

The PeopleSoft Inventory feature that updates tables based on count result input, regardless of how the count was created or the data collected.

Physical Inventory Process

In PeopleSoft Asset Management, the process by which you extract asset data from the Asset Management database to load into your bar code scanning device. You then scan the assets and load the data gathered during the actual physical inventory into Asset Management, enabling you to generate physical inventory results for review. You perform matching and generate transactions to reconcile the data in Asset Management with the results of your physical inventory.

PIA

PeopleSoft Internet Architecture. This is the fundamental architecture on which PeopleSoft 8 applications are constructed, consisting of an RDBMS, an application server, a web server, and a browser.

Piece Work

Method of compensating time reporters based on units completed rather than hours worked

PIN

Technical term for an element. In PeopleSoft Global Payroll, PIN is often referred to in the online object names and within the batch code. PIN stands for Pay Item Name.

Placeholder

A temporary location designator in an engineering bill of material for a component item that has yet to exist. These temporary placeholders have to change into approved items before transferring engineering bills of material (EBOM) to manufacturing bills of material (MBOM).

Plan Administrator

The person selected by the employer to perform the administration of a plan under PeopleSoft Pension Administration.

Plan Eligibility

The PeopleSoft Pension Administration function that uses job data to determine whether an employee may participate in a pension plan. An employee can be eligible based on job data but not be participating because of an unmet service or age requirement.

Plan Type

A unique ranges of codes used during payroll calculation to determine deduction processing rules. See also Benefit Plan Type.

Plan Year

The annual period that a pension plan uses to measure service, earnings, and benefits. Generally, the pension plan year will match the fiscal year of the plan sponsor.

Planning Item

A non-inventory item designated as being used for planning purposes only. It can represent a family or group of inventory items. It can have a planning bill of material or planning routing, and can exist as a component on a planning bill of material. A planning item cannot be specified on a production or engineering BOM or routing, nor used as a component in production. Quantity on hand will never be maintained.

Planning Level

The level ona dimension's tree used for planning. Typically, a customer will choose not to plan at the lowest level of available detail, such as the individual product level. Instead, the individual products are mapped to their corresponding product group and the planning is done at the product group level.

Planning Targets

The amount the budget must equal, such as a budget spending limit or cap for expenses where users can not exceed the amount. Planning Targets are presented at a summary level. This term is interchangeable with Spending Limits.

Planning Target Tolerance

The percentage and/or amount a user can be over or under the planned budget target.

Planning Target Control

Values set at the user role level, including no control, active control, and passive control. For more information see no control, active control, and passive control.

Planning Task

Any activity in PeopleSoft Planning that creates a schedule.

PODS (PeopleSoft Operations Data Stores)

See Data Warehouse Tables

POI. Personnel Office Identifier.

Also known as Submitting Office Number (SON). These are codes assigned by the OPM to the office(s) delegated authority within an agency to process personnel actions on Federal employees.

Pointers

A pointer is an "address" of a driver quantity, or value, within the Enterprise Performance Management product line. Pointers are used as a means of defining where driver quantities exist in tables that reside in the PeopleSoft Enterprise Warehouse. Pointers enable you to extract values from any location in the warehouse and then use these values as driver quantities. There are three different kinds of pointers: explicit, implicit, and multidimensional.

Policy Control Group

In PeopleSoft Inventory Planning, a feature for setting up order quantity, safety stock, reorder quantity, and maximum and minimum policies. The **Control Group** is assigned to a set of **Planning Item**. The policies of the associated planning items can be set explicitly or defaulted from the policies on the policy control group.

Policy Generation

In PeopleSoft Inventory Planning, a set of run options used to control the functions and behavior of the Policy Generation program.

Policy Item

An item record which is related to a location and for which **Inventory Policy** is held. A policy set, **Planning Item** ID, and Location ID uniquely identify a policy item. The combination of an item and a location is called a stockkeeping unit.

Policy Set

Defines a set of the items for which **Inventory Policy** is to be calculated. Each policy set is assigned a unique ID and includes information that defines, for example, the associated **Forecast View**, time periods, and planning horizon.

Policy Simulation

In PeopleSoft Inventory Planning, a feature that simulates the effects of various stocking scenarios, compares current policy with simulated policy, and determines the best inventory investment strategy.

Population

A Population is defined on top of DataMaps created using Enterprise Warehouse metadata. The Population builder allows you to easily format a SQL statement, using filters, to select

rows from one or more tables joined together in a DataMap. You can view the results of a Population directly from the browser.

Pop Up Payment Option

A variation on a joint and survivor payment option under which the benefit payable to the participant is increased if the beneficiary should die prior to the participant.

Portal

A portal is a web site that helps you navigate to other web-based applications and content. Users often consider a portal their "entry point"—the place they typically visit first after launching their web browser.

Portal Registry

The portal registry is a tree-like structure in which content references are organized, classified, and registered. It is a central repository that defines both the structure and content of a portal through a hierarchical, tree-like structure of "folders" useful for organizing and securing content references.

Portal Registry API

The Registry API is provided for accessing each portal registry from PeopleCode, COM, Java, or C programs. Providing the same kind of registry management capability as the online administration pages, it can be used by external systems to update the registry to reflect changes in the content reference URL, taxonomy, and effective dates. The Registry API is fully described in the PeopleCode documentation.

Portal Servlet

A Java servlet that runs on a web server. The portal servlet intercepts user requests for content, retrieves content, and builds a single HTML document to be displayed in the user's browser.

Portal Solutions

Portal Solutions are separate product offerings from PeopleSoft that consist of pre-built, packaged solutions focused at different audiences (customers, suppliers, and employees). Because they are both pre-built, supported application products, Portal Solutions can be deployed swiftly and easily, saving significant resources when compared to other custom-built solutions.

Position

The officially assigned duties and responsibilities that make up the work performed by an employee. Positions are linked to Job Codes, which can be considered the electronic version of the Position Description. There can be a many-to-one relationship between the Position and Job Code.

Position Budgeting

The budget amounts (salary, benefits, and earnings) associated with positions within an organization. Position budgeting can be calculated based on position information loaded from a human resource system. Position budgets are used to generate line-item budgets for personnel costs. PeopleSoft Budgeting-specific.

Position Change

A move by an employee to another position during the employee's continuous service under the same appointment within the same agency.

Position Date Created

Date the position was created for use in the agency.

Position Description (PD)

In accordance with OPM guidelines, an official description, authorized and approved by an agency official, describing duties and responsibilities to be performed. Position classification standards are used to describe the work, classify the work components by occupational series, and factors (e.g. supervisory control, scope, complexity, competencies required) are used to determine the grade level (i.e., salary range) for the position.

Position Description Number

A number assigned to identify various types of Position Descriptions.

Position Description Required

Identifies those positions for which a position description must be maintained.

Position Number

A number that identifies an authorized Position.

Positive Input

Data such as hours worked or a bonus amount entered for elements that change each pay period. Positive input can be entered manually, generated by the system, or received from other applications.

Positive Task Reporting

A method of time reporting in which all required task elements must be provided (see Time Reporting)

Positive Time Reporting

A method of time reporting in which all elements of time must be provided (see Time Reporting)

Post Differential Percent

Additional compensation that may be paid to certain employees who work in Guam or the Northern Mariana Islands.

Post Differential, Non-Foreign

A differential payable to an employee at a location in a non-foreign area if conditions of environment differ substantially from conditions of environment in the contiguous United States and warrant its payment as a recruitment incentive.

Post-56 Military Deposit

The OPM provides guidelines to Federal agencies on how to calculate and process these voluntary employee deductions from pay toward the employee's current retirement fund for those periods of eligible military service.

Posting

In PeopleSoft the process by which accounting entries are created or updated based on user transaction input and accounting entry templates. In PeopleSoft Receivables, posting is also known as Receivable Update.

These posted accounting entries in the feeder systems,such as accounts receivable or accounts payable, must be further processed by the Journal Generator to create journal entries before posting to the General Ledger occurs.

Post-Tax Deductions

See After-Tax Deductions

Pre-encumbrance

An encumbrance that occurs before an employee/employer relationship exists. You encumber funds for an employee you have on staff; you pre-encumber funds for an employee that you anticipate hiring. For example, you would pre-encumber funds for a new position that has just been approved but not filled.

Pre-encumbrance Ledger

Stores pre-encumbrance amounts. Updated by posting pre-encumbrances, such as purchase- or hiring-requisition source transactions (including journal entries).

Premium

Any additional compensation for extra hours worked, often expressed in terms of factor-above-normal-per-hour pay, such as time and a half (where one-half is premium pay), double time or triple time. Also, any additional pay provided to a time reporter based on compensation rules (see Time Administration in your *PeopleSoft Time and Labor* PeopleBook).

Prenote

A prenotification or waiting period requested by banks before processing payroll direct deposits.

Pre-Retirement Survivor Annuity (PRSA)

A benefit paid to a beneficiary if a pension plan participant dies before commencing benefits. Qualified plans must offer a pre-retirement survivor annuity, although the employee can be required to pay for the coverage with a reduction in the benefit.

Pre-Tax Deductions

See Before-Tax Deductions

Previous Day's Close

The previous trading day's last reported trade.

Price Break

Defined in PeopleSoft Order Management, and linked with price sets, a price break defines the actual adjustments that are made to the list price. They are valid only within a time frame you establish.

Price Rule

Used in conjunction with price sets, rules are essentially a decision tree that establishes the search order the system uses in reviewing sets for a match on the variables they reference for price adjustments.

Price Set

Linked with price breaks, a price set specifies the parameters for your price adjustment. It establishes selection criteria, determines whether the break is based on quantity or price, and defines how the adjustment is applied.

Price Source

A service provider or publication that reports the trading activity for a stock traded on stock exchanges. Examples include Wall Street Journal and Bloomberg.

Primary event code

Primary event codes, also called purpose codes, specify the status of the transaction: whether it's a new transaction, a cancellation, a duplicate, a status request, and so on. Every transaction has a primary event code assigned to it.

Primary Insurance Amount (PIA)

The benefit amount calculated under the Social Security benefit formula.

Primary scroll record

Primary scroll records are the principal SQL table or view associated with a page scroll level. A primary scroll record uniquely identifies a scroll level in the context of its page: each scroll level can have only one primary scroll record; and the same primary scroll record cannot occur on more than one scroll at the same level of the page.

Prior Period

In Time and Labor, any payroll period before the current one.

Prior Period Adjustment

A change or correction to previously reported time or task information, or an insertion of time or task information. Often requires the original report to be offset (reversed) and the correct information to be recorded. (see Time Reporting)

Priority Rank

The numeric value assigned to inventory **Demand Priority Rules**. The lower the number, the higher priority of the rule and the orders matching that rule.

Private

A tracking method used by a privately held company to track their daily prices. The Board of Directors typically establishes a price for a period of time. Stock of a privately held company is not traded on an exchange.

Private Views

User-defined views available only to the user who created them. For more information, *see* Budget Views.

Process

See Batch Processes.

Process Definition

Process Definitions are created in the Process Scheduler Manager pages to define each specific run request. A Process Definition is comprised of a variety of variables including panels associated with a request, Process Groups, output parameters, run locations, and many more.

Process Group

Used to associate specific Process Definitions with a Class Profile in Security Administrator. This allows administrators to restrict an operator's ability to initiate requests.

Process Instance

A unique number that identifies each process request. This value is automatically incremented and assigned to each requested process when the process is submitted to run.

Process Job

Multiple Process Definitions can be logically linked into a job request to process each request serially or in parallel, and optionally initiate subsequent processes based on the return code from each prior request.

Process List

The set of instructions the system uses during a payroll process to determine which elements to resolve. A process list is comprised of sections that identify the sets of elements to be resolved. You build process lists and attach them to calendars.

Process List Manager

The program used during batch processing that reads the Process List and calls the PIN Manager to resolve elements on the list.

Process Request

A single "run request," such as an SQR, a COBOL program, or a Crystal report that you run through Process Scheduler.

Process Run Control

A PeopleTools variable used to retain Process Scheduler-defined values needed at runtime for all requests referencing a run control ID. This is not to be confused with application run controls, which may be defined with the same run control ID, but only contain information specific to a given application process request.

Process Scheduler

A PeopleTool that performs tasks behind the scenes of your application. It can run several kinds of processes, such as COBOL, SQR, and Application Engine programs. You can schedule processes to run on a regular schedule or at your request. Processes can run on your workstation or on a server.

Process Scheduler Server Agent

The server-based program (PTPUPRCS) that manages the selection, validation, and initiation of all queued requests for each defined server within your batch environment (Process Scheduler).

Process Type

A global process definition under which related process definitions are grouped. This allows for easy maintenance of Process Definitions that share common parameters.

Processing group

In order to partition application processing between the client and the application server, it is necessary to define units that, as a whole, run in one location or the other. We call these units processing groups.

Processing groups can encompass one or more PeopleCode events. Some processing groups can run either on the client or on the application server, such as Component Build and Component Save.

Product

A commodity primarily defined in PeopleSoft Order Management. It may be: 1) The Order Management view of an inventory item that has attributes the same as or different from those of its inventory counterpart. 2) A commodity that is not a stocked inventory item such as a product kit or a service. 3) A tangible commodity that is drop shipped from another vendor and is never stocked in inventory.

Product Alternate

Alternative products that can replace the product ordered when it's out of stock or a problem with a particular product temporarily prevents shipment.

Product Catalog

A configurable list of available products that may be of interest to a specific customer. You can create two types of catalogs—inclusive catalogs that contain all the products you want made available to a customer and exclusive catalogs that contain the products you want to withhold from your customer. By attaching product catalogs to a Sold To customer, you define the products they can buy.

Product Definitions

This support module describes how other support modules process the instruments that belong to a particular product. For example, different products have different cash flow characteristics and may be stratified differently, or may react differently to changes in interest rates. This module enables you to specify each of these actions for each instrument.

Product Kit

A commodity that consists of a fixed set of components that are sold together. It appears as a single line on an order, but is represented by multiple lines on shipping documents. Product kits may comprise inventory items, non-stockable products, or a combination of both.

Product Kit Component

A commodity that is part of a product kit. It may be an inventory item or a non-stockable product such as a service.

Product Pricing Model

In the financial services industry, this defines models that describe indices upon which future rates are paid or charged for an individual product.

Product Tree

A user-defined graphical representation of a company's product structure. A product tree defines how products are promoted and determines what users have authority to promote those products.

Production Maintenance Spreadsheets

A set of spreadsheets generated by an nVision process, containing production ID and production schedule data extracted from PeopleSoft Production Management. You can add and maintain production quantity data using these spreadsheets and then import the data back to Production Management.

Production Option

Effective-dated combinations of BOM codes and routing codes. You can create multiple effective-dated BOM code/routing code combinations (or production options) for an item. These combinations can be extracted to PeopleSoft Production Planning. They enable the specification of multiple production variations for an item and provide control of seasonal variations by effective date.

Production Option Cost

A cost based on a specific BOM/routing combination (also known as a production option). The Production Option Cost utility rolls up production options costs based on specific BOM/routing combinations, enabling you to cost individual production options and later to have the DataLink pass this cost to the Planning engine. If this utility isn't run, the Planning engine will use the standard item cost based solely on the primary BOM and routing instead.

Productive Time

Employee scheduled time spent performing any task for which a position was created; work performed on behalf of a business entity that is required for that entity to fulfill its business purpose. Employees doing the work they or someone else was hired to do.

Productive Unit

In Italy employers organize employees into productive units based on agreements between the unions and the employer.

Profile

A data set that you aggregate from the Enterprise Warehouse, according to the filters you specify, the Key Performance Indicators you select, and the 3rd party demographic data you include.

Profile Factor

In PeopleSoft Demand Planning, the weight index assigned to each **Forecast Period** to take into account seasonal fluctuations in demand. The factor or index typically measures the percentage of difference between the base demand and the expected actual demand in the period.

Profiles

Group of employees defined according to a list of job codes and departments. You can use these profiles to ascertain training demands within your organization based on set criteria.

Profit Manager

The Profit Manager is a set of integrated tools that enable true profitability reporting. Profit Manager features are tightly integrated with the PeopleSoft Analytic Applications and provide you with ways to ensure data integrity, edit data, and post data to the Performance Ledger table.

Project

The highest level of hierarchical organization within PeopleSoft Projects. Projects provide the structure to which activities and resources are added. Each node on a Projects tree represents a project. Projects can contain other projects as well as activities and resources. This provides a hierarchical relationship between projects and facilitates cost roll-ups.

In Enterprise Performance Management you use a Project to create or modify a Profile. A Project contains pointers to data elements that you include in a Profile.

Project

In PeopleSoft Time and Labor, a specific endeavor undertaken to achieve a specific goal. Typically, projects are approved and undertaken with level of cost, schedule, and performance already agreed upon. A project is composed of a set of tasks, each of which requires staffing, provisioning, and/or scheduling. Project progress is often measured in terms of task completion.

Project ID

The name or number by which a project is to be identified in all tables and pages.

Project Type

A user-definable grouping of projects. Project types are optional.

Projected Run Date

In PeopleSoft Demand Planning, a calculation made that projects a life volume for a period based on a calculated run rate or performance ratio.

Projection.

An estimated pension benefit calculated as of a future date or any estimated data used as the basis for such a calculation.

Promotion

- For positions under the same type job classification system and pay schedule, a promotion changes the employee to a higher grade level or makes permanent a Promotion NTE;
- When the old and new positions are under different job classification systems and pay schedules, a promotion changes the employee to a position with a higher rate of basic pay or makes permanent a Promotion NTE.

Promotion Pattern

In PeopleSoft Demand Planning, an **Event** function that enables you to apply weights to promotions across a range of **Forecast Period**.

Prompting Profile

A task profile usually used by account managers as a way of creating task profiles for employees who report task time differently by customer. For example, you might have an account manager who has fifty customers; when the account manager comes in each day to report time, the system will display all the customers, and indicate which customers it will use as a default if she doesn't manually report time.

Prorated

In Enterprise Planning and Simulation, prorated is when the computed forecast and the summarized forecast are two different versions of the statistical forecast. In addition, the forecast at the product family level can be allocated down to the individual products. Usually this allocation is done in proportion to the calculated product forecasts at that level. This version of the (statistical) forecast is called the allocated or prorated statistical forecast.

Pro-Rate Purchase

A purchase in which the number of shares to be purchased is prorated according to a specified factor. This may occur when the total number of shares to be purchased is greater than the number of shares allocated to the stock plan from the treasury.

Prorated Forecast

In PeopleSoft Demand Planning, a forecast developed by factoring the group forecast down one level at a time to make the sum of the item forecast equal to the aggregate forecast. The prorated forecast tends to be more accurate than the **Statistical Forecast**.

Proration Rule

Element that defines how you want to prorate an item. You use proration rules in numerous places—for instance you could prorate an earning, deduction, or many of the elements that make up an earning or deduction.

Proration Threshold Ratios

In PeopleSoft Demand Planning, the upper and lower ratios used as thresholds for **Reasonableness** checks when a forecast is developed using proration.

Provider

An entity, such as an insurance company, that provides one or more of the benefits your company offers. For example, Metropolitan Life Insurance Company is a provider to companies that use a Metropolitan life plan.

Proxy Person

A highly compensated executive. Corporations must include information regarding the most highly compensated executive officers in their proxy reporting.

Proxy Statement

The document that must accompany a solicitation of proxy appointment under SEC regulations. The purpose of a proxy statement is to provide shareholders with the appropriate information to make an intelligent decision.

PSADMIN

A PeopleSoft utility providing a menu interface to create, configure and administer application server domains and the Process Scheduler Server Agent (Batch Server).

PSADMIN

A PeopleSoft utility providing a menu interface to create, configure and administer application server domains and the Process Scheduler Server Agent (Batch Server).

PSAPPSRV

PSAPPSRV is the main server process running within a domain. PSSAPPSRV performs the functional requests, such as building and loading components. It also manages the memory and disk-caching for PeopleTools objects on the application server. Each PSAPPSRV process maintains its own memory and disk cache.

It provides authentication services for incoming users. For instance, it checks the PeopleSoft OPRID against the directory server or PSOPRDEFN table.

PSQCKSRV

Essentially, PSQCKSRV is a copy of the PSAPPSRV. It performs quick, read-only SQL requests. It is an optional Server Process designed to improve performance by handling items in the PSAPPSRV transaction request queue.

PSQRYSRV

Like the PSQCKSRV server process, PSQRYSRV is designed to alleviate the workload of PSAPPSRV. PSQRYSRV is designed to specifically handle all user-generated queries submitted by PeopleSoft Query (PSQED.EXE). This server process is designed to improve overall application server performance whether or not you have PSQCKSRV configured. It is specifically, and exclusively designed to process PeopleSoft Query transactions, which can be very SQL intensive.

PSSAMSRV

It processes conversational SQL transactions primarily for Application Designer.

Public Company

A company that has held an initial public offering and whose shares are traded on a stock exchange or in the over-the-counter market. Public companies are subject to periodic filing and other obligations under the federal securities laws.

Public Views

Coordinator-defined views, available to anyone using the application. For more information, *see* Budget Views.

Publish/Subscribe

Publish/Subscribe type messaging is performed with PeopleTools Application Messaging technology. You can send data from one PeopleSoft system to another in an asynchronous mode—meaning the two systems don't have to be sending and receiving at the same time. This is possible because the message transfer is accomplished through a Web server with an "http: gateway."

Pull List

Similar to a pick list, a pull list contains multiple replenishment requests, including the location, quantity, and item quantity required in a specific sorting sequence. You use pull lists in PeopleSoft Flow Production with Inventory replenishment.

Pull Ticket

A document containing the details of a single request replenishment request, including Kanban ID, item, quantity, source, and To locations. You use pull tickets in PeopleSoft Flow Production with Inventory replenishment.

Punch

Precise instances of date and time recorded for a user and measured in seconds, minutes, hours, day, month and year and time zone (see Time Reporting)

Punch Duration

Length of time between two punches in increments of hours or partial hours (see Time Reporting)

Punch Matching

Area of the application which converts paired punches to punch duration by processing rounding rules and assigning the tasks to the appropriate logical day based on rules established by the user

Punch Restriction

The facility to constrain a time reporter's ability to create a punch that deviates from the schedule (see Time Reporting)

Punch Type

A user defined classification of punches, i.e. In, Out, Start, Stop (see Time Reporting)

Purchase

The issuance or purchase of shares through a stock purchase program. The purchase is made using current contributions from a participant and any carry-forward remaining for the participant from previous purchases.

Purchase Price

The discounted price paid for the shares at the end of a purchase period.

Purchase Price Variance

A PeopleSoft Payables matching feature that compares purchase order and inventory prices for any variance in the prices.

Purge Rules

The rules that define criteria to clear data you no longer need from previous open enrollment processing cycles in PeopleSoft Benefits Administration.

Pyramiding

A computer calculation enabling an individual owner of one share of stock to use the stock-swap technique to exercise a stock option of any size without using cash. Not many corporations permit pyramiding.

Q

QDRO

See Qualified Domestic Relations Order.

QDRO Alternate Payee

A former spouse who is entitled to a portion of a participant's pension benefits as a result of a court order.

QJSA (Qualified Joint and Survivor Annuity)

A post-retirement death benefit for a spouse. Plans subject to this requirement must provide an annuity for the life of the participant with a survivor annuity for the life of the participant's spouse.

QMCSO (Qualified Medical Child Support Order)

A QMCSO is a court order that requires a group health care plan to provide benefits to the child of a participant as part of a child support arrangement on the behalf of that participant. Base Benefits enables the tracking of QMCSOs for dependents.

Qualified Domestic Relations Order (QDRO)

A court order ordering a division of a participant's pension benefits. This is normally the result of a divorce and gives a portion of the pension benefits to the former spouse.

Qualified Plan

A pension plan for which the employer can take tax deductions for contributions to the plan. Investment income of the plan trust fund is not taxable to the employer. Tax law places restrictions on the plan rules.

Qualifying Dispositions

A transaction whereby a participant sells shares acquired through a stock purchase plan two years after the grant date and one year after the purchase date.

Quality Function

A level of configuration that enables you to define the fields and attendant information that provides a base level for inspection plan and integration. Quality functions enable you to map process-specific field information into PeopleSoft Quality for identification, tracking, and analysis.

Quality Ranking Factors

Knowledge, skills, and abilities that could be expected to enhance significantly performance in a position, but are not essential for satisfactory performance. Applicants who possess such

KSAs may be ranked above those who do not, but no one may be rated ineligible solely for failure to possess such KSAs.

Quality Server

A PeopleTools-based analysis and graphing program.

Quality Step Increase (QSI)

A step increase awarded to an employee for sustained high quality performance.

Quantity Allocation Method

In PeopleSoft Inventory, the method used to determine how available quantity will be allocated to prioritized demand lines when using the online reservations page.

Quantity Precision Rules

A set of rules specifying whether item quantities for a given unit of measure are expressed as whole numbers or as decimals. Quantity precision is set at the inventory **SetID** and item-UOM levels.

Query

A set of data members that are selected from a Class catalog (provided by the Business Interlink Plug-in) as well as a generic form of Criteria. The criteria are composed of <left-hand-side> <Relational Operator> <right-hand-side> statements that can be concatenated using a set of logical operators. All operators and class catalogs are dynamically provided through the Business Interlink Plug-in.

R

Race And National Origin Code

A code that identifies the employee's basic racial and national origin category.

Range of Dates Reporting

A Time and Labor process that enables you to report a start and stop date, a time reporting code and task information for a single employee. The system transforms the information into instances of daily time based on the employee's schedule or default work schedule, replacing the scheduled time with the entered Time Reporting code and the number of scheduled hours on a day-to-day basis.

Range Penetration

In PeopleSoft Workforce Analytics, Range Penetration is the degree to which an employee's actual pay has progressed through their salary grade, and is expressed as a percentage. The calculation is:

Range penetration = (Employee Base Pay – Range Minimum)/(Range Maximum – Range Minimum).

Range Width

In PeopleSoft Workforce Analytics, the difference between the maximum and the minimum values of the pay range calculated using the following formula (and expressed as a percentage):

(Maximum – Minimum)/Minimum.

Rapid Time Entry

The process that enables you to enter daily time for single employees without the system editing your field entries. The system populates temporary tables, which are used by a batch process that reads, edits and moves the data into the appropriate time and labor tables. You cannot prompt for valid values in any of the fields, and the online system does not edit any of the data you enter against other tables.

Rate Code

Alphanumeric reference to the cost per hour or unit of time reported to a specific TRC.

Rate Code [Global Payroll]

IDs for pay components. Rate codes define rates of pay and are set up in the Comp Rate Code table. Rate codes are then used to represent pay components in pages and when you configure compensation packages in the compensation record.

Rate Code Group

A rate code group is a number of pay components (represented by rate codes) bundled into a subset of a compensation package. The rate code group is used to calculate percentage-based components that do not apply the percentage to all pay components in the compensation package. Rate code groups are constructed on the Rate Code Groups page.

Rate Code Type

Defines how the monetary value of the rate code is calculated. The compensation rate code type is defined on the Comp Rate Code table. Valid values are Flat Amount, Hourly Rate + Flat Amount, Hourly Rate, Percent, and Points.

Rate Combinations

The combination of rate types and conversion rates with account types that is linked to your budgeting model. Typical rate types are current, commercial, floating, average, and historical. Effective dates define different rates for different budget periods. There are several conversion rates for any pair of currencies including not only the current rate, but others rates such as average, historical, asking, and so on. These different types of rates are appropriate for different types of accounts.

Rates

The arrays of values used to calculate the cost of a plan to an employee. Rates can be age-graded, service-related, or general, depending upon the benefit plan type. Any number of benefit program and benefit plan combinations can use each set of rates.

In Enterprise Performance Management, a rate is determined by the user and specifies the dollar amounts to be calculated for each model. This is a financial services industry term.

Rating Model

The scale used by your company to measure competency proficiency. The default rating model is the PSCM (PeopleSoft Competency Management) Rating Model that PeopleSoft delivers with your PeopleSoft Human Resources System.

Raw Punches

See Actual Punch; typically this is distinguished from a rounded punch (see Time Reporting)

Reason Code

Reason Codes provide explanations for occurrences such as returned stock and changes to order headers, lines, or schedules.

Reason Code

A code describing employee time such as comments for sick time or travel time.

Reasonableness

In PeopleSoft Demand Planning, a technique that checks the trend and projected annual growth to make sure that a forecast is realistic. If a forecast falls outside either boundary, the system automatically adjusts it and sends a warning message.

Reassignment

Change of an employee from one position to another without promotion or change to lower grade.

Recalculate Forecast

In PeopleSoft Demand Planning, a forecasting feature that uses the existing model and its associated parameters to create a new forecast.

Recalculate VAT at Payment

Allows the VAT amount to be adjusted at the time of payment if an early payment discount is taken. This calculation option is only valid when VAT is calculated at Gross.

Receipt Cost Method

Determines how you cost receipts. Receipt cost methods include Actual, Non-Cost, and Standard.

Receipt Line

A line associated with a Receipt ID that identifies an item and quantity. If the respective tracking is activated, the lot, lot suffix, and serial number are also identified.

Receivable Update

See **Posting**

Receivables Item

An individual receivable. An item can be an invoice, credit memo, or debit memo. Items and payments combined comprise a customer's balance.

Reconciliation

Within PeopleSoft Enterprise Performance Management, reconciliation differs slightly when it is performed within the PeopleSoft Enterprise Warehouse and when it is performed within the PeopleSoft Analytic Applications.

In PeopleSoft Funds Transfer Pricing (FTP) and PeopleSoft Risk Weighted Capital (RWC), reconciliation identifies differences between Performance Ledger balances and the instrument or position balances, which are risk weighted according to the basis rules you have assigned. The first step in reconciling basis rule balances is to reconcile the individual balances for accounts, instruments, and positions. Reconciling the total balances is the second step. This means that you reconcile the difference between Account/Instrument balances, and the difference between Account/Position balances.

In the PeopleSoft Enterprise Warehouse, reconciliation is a period-end process that posts journal entries to the Performance Ledger for the discrepancies found when you reconciled the individual balances. Typically, you'll run the PF Reconciliation engine after a period to compare "to and from amounts" between tables such as REVENUE_F00 and PF_LEDGER, or the GL_LEDGER and the PF-LEDGER.

Record Date

The date a stockholder must officially own shares in order to vote at the meeting or to derive an adjustment resulting from a stock split or a stock dividend. The Board of Directors sets the Record Date.

Record Definition

A logical grouping of data elements.

Record field

Once a field is inserted into a record definition it becomes known as a Record Field within the record.

Record Group

A set of logically and functionally related control tables and views. Record groups exist for two basic reasons:

- To save you time—with Record Groups, TableSet sharing can be accomplished quickly and easily, eliminating an enormous amount of redundant data entry
- To act as a safety net—Record Groups ensure that that TableSet sharing is applied consistently across all related tables and views in your system.

Record Input VAT Flag

Within PeopleSoft Purchasing, Payables, and General Ledger, this flag indicates that you are recording input VAT on the transaction. This flag, in conjunction with the Record Output VAT Flag, is used to determine the accounting entries created for a transaction and to determine how a transaction is reported on the VAT return. For all cases within Purchasing and Payables where VAT information is being tracked on a transaction, this flag is always set to Yes. This flag is not used in Order Management, Billing, or Receivables, where it is assumed that you are always recording only output VAT, or in Employee Expenses, where it is assumed that you are always recording only input VAT.

Record Output VAT Flag

For certain transactions within PeopleSoft Purchasing, Payables, and General Ledger, it may be necessary to record both input VAT and output VAT on the same transaction. Generally, this would be a situation where the purchaser was required to self-assess VAT. Within these situations, this flag indicates that you are recording output VAT on the transaction. This flag, in conjunction with the Record Input VAT Flag, is used to determine the accounting entries created for a transaction and to determine how a transaction is reported on the VAT return. This flag is not used in Order Management, Billing, or Receivables, where it is assumed that you are always recording only output VAT, or in Employee Expenses, where it is assumed that you are always recording only input VAT.

Record Owner

The "Stockholder of Record" of the stock. This may be different from the "Beneficial Owner" of the stock.

Record Suites

Record suites are temporary tables that enable the system to track how many processes are running. These temporary tables leave the fact tables accessible for processing other jobs simultaneously without impacting your processing.

Reduction In Force (RIF)

Method used to reduce the number of government workers in an agency.

Reemployed Annuitant

An employee who has retired from Federal employment and is receiving an annuity. His/her salary is reduced by the amount of the annuity.

Reference Designators

A user-defined alphanumeric identifier that determines where a component is placed in an assembly.

Reference Transaction

In People Soft commitment control, a reference transaction is a source transaction that is referenced by a higher-level (and usually later) source transaction, in order to automatically reverse all or part of the referenced transaction's budget-checked amount. This avoids duplicate postings during the sequential entry of the transaction at different commitment levels. For example, the amount of an encumbrance transaction (such as a purchase order) will, when checked and recorded against a budget, cause the system to concurrently reference and relieve all or part of the amount of a corresponding pre-encumbrance transaction, such as a purchase requisition.

Referential Integrity

Issues that occur when an update to an instance of one object invalidates one or more instances in a related object. In other words, when you make a change to one area of the application, referential integrity makes sure the changes do not adversely affect another area of the application.

Refresh Time

The process that retrieves the appropriate current version of objects related to employee time (such as task profiles or work schedules) and associates them with that time.

Region Codes

Regions may or may not be physical entities, they may simply be another way to geographically categorize an area. When a region does represent a physical entity, the region code has the same characteristics as a business, that is, an address and a language spoken.

Register of Separations and Transfers (ROST)

The ROST is a regulatory compliance document used by federal agencies to summarize the information in an employee's Individual Retirement Record (IRR). The ROST is a one-page cover sheet that accompanies a batch of IRRs being submitted to the Office of Personnel Management (OPM) at the time of an employee's separation from a federal agency. Employees covered by the CSRS retirement plan require SF-2807. Employees covered by the FERS retirement plan require SF-3103.

Registration

The name or names that appear on the stock certificate to indicate who owns the stock.

Registration Statement

The document that must be filed to permit registration of an issue of securities under the Securities Act of 1933. A major component of the registration statement is the prospectus that is to be supplied to prospective purchasers of securities.

Regression Analysis

A statistical technique that determines the relationship between two or more variables. Regression predicts the value of one variable (the dependent variable) based on one or more independent variables.

Regular Base Compensation

In PeopleSoft Workforce Analytics, the annualized, quoted, compensation rate for a job. Consists of fixed compensation, does not include variable compensation.

Regular Time

An employee's normal (scheduled/shift) work hours.

Regular Time

In PeopleSoft Workforce Analytics, an employee's normal (scheduled/shift) work hours.

Regulation T

Federal Reserve Board regulations governing the extension of credit by brokers or dealers, including their participation in same-day sale transactions and sell to cover exercise.

Regulatory Region

The Regulatory Region functionality in PeopleSoft HRMS is designed for use in performing regulatory and regional edits. You'll use Regulatory Region to drive PeopleCode edits, perform set processing, and control what codes and values the operator sees. A Regulatory Region can be any country (or province or state) where there are specific laws and regulations addressed by functionality in PeopleSoft HRMS.

In Enterprise Performance Management, a Regulatory Region is a region with a common regulatory framework; such as a country (CAN for Canada), or a smaller state or provincial entity (CANBC for British Columbia).

Related Education

Education above the high school level that has equipped the applicant with the KSAs to perform successfully the duties of the position being filled. Education may relate to the duties of a specific position of to the occupation, but must be appropriate for the position being filled.

Release

An industry standard term associated with the lifting of a company's Repurchase Option from a portion or all shares from a Restricted Stock Award (RSA). RSA's are subject to release schedules, similar to vesting schedules.

Relevant Constraint

A constraint PeopleSoft Planning considers when it calculates a score and when it optimizes the schedule. See also **Scorecard** and **Optimize**.

Reloads

Some stock option plans provide for the grant of a "reload" stock option in connection with stock option exercises, typically by means of stock swaps. A reload option feature provides that upon a stock exercise, the employee will receive an automatic grant of a new stock option at the then-current fair market for the shares that they exercised or for the shares that they used to swap.

Remark Codes

Codes that cause the printing of pre-set text passages on a notice of action form. Some passages are general purpose and others are specific to the personnel action being processed.

Remit From Customer

A customer who is responsible for payments billed to other customers. During cash application, it's useful to look at open items for the Remit From group.

Remittance Worksheet

A work space in PeopleSoft Receivables used to select drafts for remittance to the bank.

Reorder Point

The identifier that automatically locates a replenishment need for an inventory item. When the physical quantity in a location falls below the reorder point, a replenishment request can be created.

Reorder Point Policy

In PeopleSoft Inventory Planning, a policy that determines when a replenishment order is launched for an item. The policy has several methods that include days supply, lead time demand, and **Fixed Quantity**.

Replacement Option

The "new" "replacement" stock option that will replace the original stock option. This option will have a grant price lower than the original stock option.

Replenish

A process that indicates when items need to be resupplied from external sources. In PeopleSoft Inventory, the process can occur on an ad hoc basis or at predefined reorder points.

Replenishment Request

In PeopleSoft Flow Production, an online request for material made when the material is needed. You can generate replenishment requests manually or automatically using backflushing. You can communicate that request using pull lists, pull tickets, or Workflow notifications.

Report Scope

A feature that creates multiple instances of an nVision report using a single report request. Each instance contains data specific to an individual ChartField, such as a business unit or department, or for a group of ChartFields, such as all sales departments. Using Scope, each report instance can share the same layout while containing data unique to these particular ChartFields.

Reported Time

Clock time or elapsed time provided to the system by the user (see Time Reporting)

Reporting Person

An insider that is regularly considered by the SEC to have material information and policy-making authority for the corporation. These individuals are subject to the reporting requirements promulgated by Section 16 of the Securities Exchange Act of 1934. Reporting Persons typically include Directors, Officers, and shareholders with 10% holding interest in the equity of the registrant's securities.

Repricing

An agreement between the corporation and the optionee that allows the optionee to cancel an outstanding high-priced, usually "Out-of-the-Money" stock options for lower-priced options.

Repricing Election

Eligible optionees can choose (elect) to accept the corporation's repricing offer or choose to decline the offer.

Repurchase

The reacquisition of shares of stock from an individual by a corporation. This usually occurs when an individual fails to meet the vesting requirements on a RSA or option that is exercised before it vested. The corporation might pay the original cost of the shares to the individual or the fair market value of the shares at the time of repurchase.

Repurchase Option

An irrevocable, exclusive option to repurchase up to the number of shares that constitute Unreleased Shares at the original purchase price per share. The Company shall exercise said option. The repurchase of outstanding shares is regulated under the laws of all states (except Massachusetts). Under some laws, as under the Model Business Corporation Act, the repurchase is prohibited unless the corporation remains solvent, in both the equitable and bankruptcy senses of insolvency and after taking any liquidation preferences of other outstanding stock into account.

Repurchase Right

A company's contractual right to buy back from an employee any stock resulting from the exercise of the option. The buy back can be at fair market value, book value, or the original purchase price.

Reservation Method

The method used to reserve soft reservation items — either batch COBOL reservations or on-line allocation and reservation.

Reserved

A flag indicating that the inventory item is reserved for stock fulfillment in the inventory business unit.

Reserved Orders

Orders that have been reserved against on-hand available quantity at the business unit-item level. Reserved orders are found in the DEMAND_INV table.

Reset

In PeopleSoft Demand Planning, a function of the **Forecast Calculation Process** that determines which forecast model will produce the best forecast, meaning the model with the lowest ratio of error.

Resolution

An activity that closes or partially closes a deduction, such as matching it to a deduction authorization, writing it off, or sending it back to PeopleSoft Receivables.

Resolution Entry Type

Code that identifies how to process activities for items in PeopleSoft Deduction Management and how to create accounting entries.

Resolution Method

A set of rules that defines how to automatically match or write-off deductions in PeopleSoft Deduction Management.

Resolution Worksheet

The workspace in which deduction items are paired with offset items and resolved or written off in PeopleSoft Deduction Management.

Resource

In PeopleSoft Manufacturing, any crews, machines, and tools that can optionally be used at work centers to complete tasks. In PeopleSoft Performance Measurement, any homogeneous grouping of general ledger line items.

Resources

Resources are the economic elements that are required to perform the activities associated with your business. Resources are consumed in the performance of these activities, and thus denote operating costs. In PeopleSoft Activity-Based Management, resources are typically regarded as the groupings of one or more general ledger accounts. In a service business, resources might include salaries, office rentals, and costs of capital such as information systems, depreciation, real estate taxes, and other associated costs.

Resource Amount

The monetary amount of a single, specific resource transaction. The Resource Amount maps to the Posted Total Amount when posted to the general ledger.

Resource Category

A field for defining individual resource types more specifically. For example, if you have a resource type of labor but want to break it down further for tracking purposes, you might define resource categories of architect hours, carpenter hours, plumber hours, and electrician hours. Resource categories are optional.

Resource Driver

In Activity-Based Management, a Resource Driver defines the quantity of resources used by an activity.

Resource Group

A category of resource types. You can define relationships between the resource types within a resource group to facilitate analysis of project costs. For example, if you had resource types for standard labor and overtime labor, you could group them together in a resource group to calculate total labor.

Resource Planning

In PeopleSoft Activity-Based Management, Resource Planning focuses on resources allocations that create expected results like driver rates and cost object costs.

Resource Quantity

A field on each resource transaction that identifies nonmonetary quantity. For example, on a resource line for 12 ball valves the quantity would be 12.

Resource Source

A field on each resource transaction that identifies the system in which the cost originated. For example, PeopleSoft Payables would be the resource source for a resource transaction created from a voucher in that system.

Resource Subcategory

A field for defining individual resource types and categories more specifically. For example, if you have a resource type of labor, and resource categories of architect hours, carpenter hours, and plumber hours, you might want resource subcategories of regular hours and overtime hours. Resource subcategories are optional.

Resource Supplied

An attribute that enables you define a resource as committed or flexible. A committed resource is one that will not likely change in the short term. A flexible resource is more likely to change within the short term.

Resource Transaction

An individual cost line within PeopleSoft Projects. It is through resource transactions that individual costs and types of costs are tracked. Each resource transaction contains a cost and a quantity and as many identifiers of that cost as necessary. Resource transactions are created when you receive information from other systems, run allocations with project resources as the target, or perform internal transactions such as billing, project closure, or adjustments.

Resource Type

The resource transaction field in PeopleSoft Projects that identifies the resource associated with a given cost. Resource types may be very general or very specific depending on your needs; they are used in conjunction with resource categories, resource subcategories, and resource groups.

Resource Use

Resource Use defines the behavior of a resource within PeopleSoft Enterprise Performance Management. An intermediate resource is a grouping of general ledger line items that may be allocated to another intermediate resource or to a primary resource.

Restricted Punch

A punch which is not accepted because it occurs outside of the predefined number of hours and minutes before or after a scheduled (Understanding Time Collecting Device)

Restricted Securities

Securities issued privately by the company, without the benefit of a registration statement. Restricted securities are subject to a holding period before they can be sold under Rule 144.

Restricted Stock Awards (RSA)

An award of shares of stock to an individual, typically granted at the par value or for no consideration. The shares are awarded on the basis of some future performance goal, either the passage of time (vesting) or the attainment of a specific goal. When the goal is achieved, the vesting occurs. The individual, typically, has all other shareholder rights over these shares such as, voting and dividend rights. The shares are issued in the name of the individual at the time of the award and are held in escrow until vesting occurs. If an employee terminates prior to the vesting of the shares then the company normally repurchases the unvested shares.

Retained Grade Effective Date

Date employee became eligible or began receiving a retained grade and pay.

Retained Grade Expiration Date

Expiration date of an employee's retained grade and pay.

Retest Date

In PeopleSoft Inventory, the date a lot should be inspected to determine whether it is still acceptable for fulfillment or consumption. (Retest Date = Creation Date + Retest Lead Time)

Retirement

Types of retirement are:

- Mandatory Retirement.
- Disability Retirement.
- Voluntary Retirement.
- Special Option Retirement.
- ILIA (In Lieu of Involuntary Action) Retirement.

Retirement Coverage Code

A code used to denote an employee's retirement coverage. The major ones include the following:

- Civil Service (CSRS)
- Federal Employees Retirement System (FERS) and FICA
- Foreign Service (FS)

- CSRS Offset
- CSRS - Special (for LEOs)
- FERS and FICA - Special (for LEOs)
- Social Security System
- None

Retroactive Benefits/Deductions

Deductions taken or benefits granted due to a recalculation of previous benefits and deductions. Late or modified union contracts, late paperwork, and delays in benefit enrollment processing may all result in a need for benefit/deduction recalculation.

Return Type Code

A designator on returned material authorizations (RMAs) that indicates what actions the return initiates. This may include replacement of the product or the creation of a credit memo in PeopleSoft Billing.

Reverse Split

A reduction in the number of outstanding shares of a corporation's stock, with a corresponding increase in the stock's value.

Reversionary Annuity

A form of pension payment where the retiree foregoes all benefit during his or her lifetime so that the entire benefit is paid as an annuity to a beneficiary after the retiree's death. If the beneficiary predeceases the retiree, the benefit is forfeited.

RIDDOR (Reporting of Injuries, Diseases, and Dangerous Occurrences Regulations)

Health and safety regulations in the United Kingdom requiring employers to report certain types of health and safety incidents to the Health and Safety Executive (HSE).

Rider

A special court-ordered or regulatory provision that may be applied to an enrollment to expand or limit any dependent or beneficiary coverage.

Risk Function

In Risk Weighted Capital, this is a user-defined formula that the system uses to derive risk weights.

Risk RuleSet

Used to assign a number of rules to a basis, for processing by the PeopleSoft Risk Weighted Capital Application. Used to group together a number of rules that frequently apply to the same type of balance.

Risk Type

In Risk Weighted Capital, this defines the types of risk associated with your business or activity. For example catastrophic, credit, legal, operational, regulatory, foreign exchange, market, interest rate.

Risk Weight

In Risk Weighted Capital, the risk weight is assigned by risk type, and is used to calculate capital allocation or normalized loss for the account or activity.

Risk Weighted Capital (RWC)

See PeopleSoft Risk Weighted Capital

RIZIV Code

This code is for Belgian employers to track the Federal Institute for Illness and Disability Insurance category.

ROE (Record of Employment) Reason Codes

ROE codes are defined by the Canadian government for employers to record employment actions such as Return to School or Pregnancy Leave.

ROLAP (Relational Online Analytical Processing)

ROLAP refers to the analytical processing and analysis of a relational Data Mart cube. ROLAP, is a form of OLAP that leverages the power and flexibility of relational databases.

Role

A role consists of a designated set of tasks, competencies and accomplishments required for a job code or a position.

Role user

A PeopleSoft Workflow user. A person's Role User ID serves much the same purpose as their Operator ID does in other parts of the system. It allows the system to uniquely identify the user and to determine what data the user has access to.

PeopleSoft Workflow uses Role User IDs rather than Operator IDs because it needs different kinds of user information than the rest of the system does. Specifically, it needs to know how to route work items to the user---an email address, for example---and what role the user plays in the workflow. Plus, you can include role users in your workflow who aren't PeopleSoft application users and who don't have Operator IDs.

Roles

Roles describe how people fit into the workflow. A role is a class of users who perform the same type of work, such as clerks or managers. Your business rules typically specify what user role needs to do an activity.

Roll Forward

In commitment control, rolling budget balances forward from the budget ledger you are closing (the source budget ledger) into the new (target) budget ledger.

In PeopleSoft Enterprise Planning and Production Planning, a utility that moves tasks from the past to a valid point in the future using time periods rather than fixed dates.

Roll Up

The act of totaling sums based on information tree hierarchies. You can roll up data for any group of details that you have defined as dependent with the Tree Manager.

Rolling Plan

An ESPP offering period where the purchase date is measured from the offer start date. If at the purchase date, the current stock price is lower than the last stock purchase price, you may elect to reset your employees to the new lower purchase price. The offering period is now based of the new purchase date.

Rollup

In PeopleSoft Demand Planning, the process of adding up the demand and forecast **Array** from one level to the next from child to parent. Information such as caption, description, and unit of measure can also be rolled up. The process is also referred to as summarization.

Rounded Punch

A punch that has a company's rounding requirements applied to it (see Time Administration)

Rounding Rule

Defines a rounding rule. You use rounding rules in numerous places—for instance you could round an earning, deduction, or many of the elements that make up an earning or deduction.

Routing (Manufacturing and Engineering)

A set of information detailing the method to manufacture a particular item. It consists of sequentially numbered operations that reference the task to be performed, the work center in which the task is to be performed, the resources to be used, and the time required to complete the task. Engineering Routings differ from Manufacturing Routings in that they are not visible within Production Planning or Production Management and are isolated from Manufacturing.

Routing Option

In PeopleSoft Planning, a valid method for replenishing supply for an item. There are two types of routing options: build options and purchase options. An item may have more than one routing option.

Routing Transit Number (RTN)

A number that identifies the financial institution to which an electronic payment should be sent for deposit.

Routings

Routings connect the activities in the workflow. They are the system's means of moving information from one place to another, from one step to the next. Routings specify where the information goes and what form it takes—email message, electronic form, or worklist entry.

RSZ (Rijksdienst Sociale Zekerheid) Category Codes

These government defined Social Insurance category codes are used to maintain social security records for your Belgian employees. RSZ Categories are associated with a Contract Type, Statute and Substitute for Claeys Formula calculations.

Rule

Representation of a company’s compensation, task allocation, or exception requirements (see Creating Rule)

Rule 10b-5

A SEC rule that prohibits trading by insiders on material non-public information. This is also the rule under which a company may be sued for false or misleading disclosure.

Rule 144

A SEC rule that applies to public re-sales of restricted securities as well as all sales by affiliates. The requirements include (1) current public information about the issuer, (2) a one-year holding period for "Restricted Securities," (3) unsolicited brokers' transactions, (4) an amount limitation. the greater of 1% of the outstanding stock or the average weekly trading volume may be sold during any three-month period, and (5) a Form 144 filing.

Rule Actions

Functions that can be used in the creation and application of a rule (see Time Administration)

Rule Elements

Customer defined pieces of information which are passed to Time Administration in order to apply and evaluate rules (see Time Administration)

Rule Period

A Time & Labor period used in the evaluation and application of a rule (see Time Administration).

Rule Program

Specifies the set of rules the Time Administration process will execute and the order in which it will execute the rules.

Rule Results

Net effect of the application of a rule; for instance, the creation of time, initiation of workflow, modification of reported tasks (see Time Administration)

Rule Templates

Templates used to quickly create a variety of rules for the Time Administration program to execute when processing reported and/or scheduled time. Some examples are. compensation rules for overtime and holidays, notification rules for irregular attendance, and rules for just about any other time-reporting situation that requires special processing.

Rules

Rules are your company's business practices captured in software. Rules determine what activities are required to process your business data.

Rule Set

Rule Sets enable you to apply basis rules to your PeopleSoft Analytic Application in the sequence that you prefer. This is particularly helpful if there are multiple basis rules for the same account node, product node, or position source code. Rule Sets can control the execution sequence of your rule combinations, filter combinations, or both. The first occurrence on the node will be applied and any other occurrence will be ignored. Rule Sets are also used with the Data Manager, and with the Currency Conversion engine.

Rules/Time Administration

A physical implementation or execution of a company’s compensation, exception and task rules (see Time Administration)

Run Control

A run control is a type of online page that is used to begin a process, such as the batch processing of a payroll run. Run control pages generally start some type of program that manipulates data in some way.

Run Control ID

A unique ID to associate each operator with their own run control table entries. Process Scheduler.

Run ID

Code that uniquely identifies a Run Control for batch processes.

RWC (Risk Weighted Capital)

See PeopleSoft Risk Weighted Capital

S

Safety Stock Policy

In PeopleSoft Inventory Planning, a policy that determines how safety stock quantities are calculated for an item. The policy has several methods that include days supply and percentage of demand fill.

Salary

Rate of compensation received by an employee.

Salary Group

Part of a group of defaults assigned to job codes. A salary group may include items such as steps and grades dependent on individual company parameters.

Salary Plan

A plan of salary defaults, grades, and step components

Salary Step Components

Pay components assigned to a salary step by entering the corresponding rate codes on the Salary Step Components page.

Salvage Value

An estimate of the amount of money one might receive upon selling an asset once that asset reaches its useful life. Salvage value is used in several depreciation calculations, including Straight Line.

Sales Order Rebate and Penalties

Rebates or penalties that are calculated against sales orders independent of **Buying Agreement**.

Sales Person

A required field used in PeopleSoft Receivables, Billing, Order Management, and Deduction Management when working with items. Each item must be assigned to a credit analyst. If no

credit analyst is assigned to an item, the credit analyst assigned to the customer is used as the default.

Sales UOM

The only units of measure that can be referenced on sales orders and quotes. You establish them on the Product Attributes by UOM page.

Same-Day Sale

An exercise and sale occurring on the same day. The exercise of the option and sale of the underlying shares take place simultaneously. The broker uses the proceeds of the sale to pay the company the exercise price and any tax withholding and the optionee the net cash (less any brokerage commission/fees).

Sample Method

One method of entering characteristic readings for a quality control plan. Using this method, for one given control plan you inspect all the characteristics for the first sample, then all the characteristics for the next sample and so on.

Scale

On a Goals Matrix (In PeopleSoft Workforce Analytics), a scale that defines the lowest, middle, and highest levels of performance needed to achieve associated minimum, midpoint, and maximum levels of compensation pay out. These are referred to as the Threshold, Target, and Stretch levels, respectively. The scale can be used to standardize multiple performance goals to a common scale of measurement.

Scenario

A scenario is a particular outcome you are analyzing when you run in Scenario Manager. Scenarios enable you to study various changes in organization models you created. For each parent and child model you want to study, you create a scenario ID that you use with all run controls.

There are two types of scenarios defined in the Scenario Manager: Historical and Forecast. In the case of an Historical Scenario all future looking fields will be display only and the Scenario Manager component serves as a wrapper to run any analytic engines. In the case of a Forecast Scenario, the Scenario Manager refers to all the business rules, forecasts, and economic assumptions that make up the scenario.

In PeopleSoft Budgeting a scenario is a ChartField used in PeopleSoft Budgeting to identify different budget iterations that use different assumptions.

Schedule

Specific task, date, and time to be worked by a Time Reporter (see Scheduling)

Schedule 13D or 13G

Disclosure forms required to be filed with the SEC and the company by a shareholder (or shareholders) that own(s) more than 5% of a public company. Schedule 13G is a short-form version of the 13D and may generally (but not always) be used only by institutional investors.

Schedule Group

A category of employees or employee groups associated for purposes of time scheduling.

Schedule Line Number

The line associated with an Order ID. The schedule line identifies an item and scheduled ship quantity that may be different from the requested quantity due to item availability.

Schedule Number

A number identifying the salary table form that an employee's pay is computed. Also has a second meaning related to the Payment Voucher processing for the ECS.

Schedule Reconciliation

In PeopleSoft Payables, the process of reconciling scheduled payments by Payment Schedule ID. Schedule Reconciliation helps U.S. federal agencies meet their requirement to schedule or group together payment orders for submission to the Treasury Disbursing Office.

Schedule Template

An ordered pattern of workday(s) and/or off day(s) used in scheduling (see Scheduling)

Schedule Type

In PeopleSoft Payables, an indicator of the nature of items purchased with a Payment Schedule**.**

Scheduled Punch.

A time reporter’s expected punch (see Scheduling)

Scheduling

A function of PeopleSoft Time and Labor and PeopleSoft Global Payroll that enables you to create work schedules and assign them to employees.

Scorecard

A weighted sum of constraint violations in a schedule that evaluates the schedule's validity (that is, acceptability). The score is calculated by adding the value for each relevant violated constraint. See also **Penalty, Weight** and **Relevant Constraint.**

In PeopleSoft Balanced Scorecard, views of a strategy tree’s components and Key Performance Indicators with red, yellow, or green scores that show its assessments.

Scrap

Any material outside of specifications and possessing characteristics that make rework impractical.

SearchIndex

A set of objects that give the programmer the ability to create, delete, insert, and update a search index and the items within it. Search index items contain a set of statistics about the document that has been indexed (keywords, number of occurrences, proximity to other words, and so on) as well as a key that can be used to point to the document (a URL, database key, or file path).

SearchQuery

A set of objects that allow the programmer to pass a query string and operators to the search engine and receive a set of matching results with keys to the source documents from the search index in return.

Seasonal Index

In PeopleSoft Demand Planning, measures the amount by which a forecasting period is influenced by seasonal effects. The index typically measures the percentage of difference between the base demand in the period and the expected actual demand in the period. An index of 100 indicates an average period in a seasonal cycle.

Seasonal Profile

In PeopleSoft Demand Planning,identifies the weight index assigned to a forecast time period to take in account seasonal fluctuations in the demand.

Seasonality Group

In PeopleSoft Demand Planning, a group of **Forecast Items** with a repetitive pattern of demand from year to year where some periods are higher than others. Typically a group of items is designated as a contributor to the seasonality group. Contributors are chosen because they are representative of the group, are stable, and have at least two years of demand history. The seasonality group profile is more stable than individual profiles of the contributors because the Aggregation process smoothes out random errors.

In Enterprise Planning and Simulation, a Seasonality group is a group of items with similar seasonal patterns. To determine if a forecast element is seasonal or nonseasonal, by averaging their history values over a year and determining where they were above and below average. Seasonality groups capture means seasonal behavior among related products.

Secondary COBRA Events

COBRA qualifying events that extend the amount of time a participant is eligible for COBRA coverage. For an event to qualify as a secondary COBRA event, it must fulfill the following qualifications: The participant must already be enrolled in COBRA coverage as a result of an initial COBRA event, the initial COBRA event must be one that is associated with a change to the employee's job status (such as a reduction in hours, termination, or retirement), and the

secondary event must be one of the COBRA event classifications that involves loss of coverage for the dependent (such as divorce, marriage of dependent, or death of employee). See COBRA and Initial COBRA Events.

Secondary event code

Secondary event codes, also called transaction codes, specify the type of transaction in detail. For example, a transaction's secondary event code could say that the transaction is a catalog order, a rush order, or a request for a sample. Not all transaction types include secondary event codes.

Section

A set of logically related elements that are to be resolved during the payroll process. You define your payroll process by creating sections and adding them to process lists. You can create four different types of sections: standard, generate positive input, sub-process, and payee.

Section 16(a)

Provision of the Securities Exchange Act of 1934 that requires company insiders to file changes in beneficial ownership of the company’s equity securities and periodic reports disclosing their holdings.

Section 16(b)

Provision of the Securities Exchange Act of 1934 that requires that any profit realized by a company insider from the purchase and sale, or sale and purchase, of the company’s equity securities within a period of less than six months must be returned to the company. It is also known as the "short-swing profit" rule.

Section 423

The Internal Revenue Code section that regulates Employee Stock Purchase Plans.

Section 83(b) Election

A tax filing within 30 days of grant that allows employees granted restricted stock to pay taxes on the exercise date, rather than the date when restrictions lapse. If an employee files the election, taxes are based on the fair market value on the exercise date, with any future appreciation taxed as a capital gain. If the employee does not file an election, taxes are based on the fair market value on the date the restrictions lapse, which will be higher assuming the stock has appreciated in value.

Securities Act of 1933

Often referred to as the "truth in securities" law, the act requires that investors receive financial and other significant information concerning securities being offered for public sale; and prohibits deceit, misrepresentations, and other fraud in the sale of securities.

Securities Exchange Act of 1934

The Congressional act that created the Securities and Exchange Commission. The Act empowers the SEC with broad authority over all aspects of the securities industry. This includes the power to register, regulate, and oversee brokerage firms, transfer agents, and clearing agencies as well as the nation's securities self regulatory organizations (SROs). The various stock exchanges, such as the New York Stock Exchange, and American Stock Exchange are SROs. The National Association of Securities Dealers, which operates the NASDAQ system, is also an SRO. The Act also identifies and prohibits certain types of conduct in the markets and provides the Commission with disciplinary powers over regulated entities and persons associated with them. The Act also empowers the SEC to require periodic reporting of information by companies with publicly traded securities.

Security Clearance

Security Clearances (Classified, Secret, Top Secret) are granted to employees by government agencies and are usually associated with jobs that bring employees into contact with classified government projects or sensitive technologies.

Security Event

In commitment control, events that trigger security authorization checking, such as budget entries, transfers, and adjustments; exception overrides and notifications; and inquiries.

Segmentation

You can “segment” components of pay based on such events as changes in compensation, employee status, or job changes during a pay period. For example, if an individual changes jobs in the middle of a pay period and your organization has a practice of separating components earned in the first job from those earned in the second job, you can set up your system to trigger segmentation of earnings results on the pay slip when there is a change to the job change action/reason field.

Selective Factors

Knowledge, skills, abilities or special qualifications that are in addition to the minimum requirements in a qualification standard, but are determined to be essential to perform the duties and responsibilities of a particular position. Applicants who do not meet a selective factor are ineligible for further consideration.

Selective Merge

In PeopleSoft Enterprise Warehouse the selective merge allows for an additional WHERE clause when you delete a merge.

Self-Service Application

Another name for PeopleSoft's HRMS and ERP applications accessed using a browser.

Sell to Cover Exercise

When an optionee sells a portion of the option shares to cover the exercise cost and any applicable taxes.

Seniority Pay

A premium paid for seniority or for the length of time an employee works for an organization.

Seniority Pay Components

Pay components whose rate codes are assigned to the seniority rate code class SENPAY (provided by PeopleSoft) on the Comp Rate Code page, allowing you to access the seniority pay functionality.

Seniority Rate Codes

A premium paid for seniority or for the length of time an employee works for an organization.

Separate Debit and Credit

A feature in PeopleSoft General Ledger that captures and reports in greater detail the accounting information that resides in balance sheet accounts. It shows the gross debit/credit balances in addition to the net balance for each account stored in the ledger. This feature also supports reversing—debit and reversing—credit journal entries for error correction.

Server Process

A server process is executable code that receives incoming client requests on the application server. The server process carries out a client request by making calls to a service that executes SQL against the database.

Service

A service performs a particular task of an application. Examples of services are MgrGetObj, SQLAccess, RemoteCall, and so on. When a client workstation sends a request to the application server, it sends a service name and a set of parameters, such as "MgrGetObject + parameters". The application server associates the service request with the appropriate server process to complete the transaction.

Service

The PeopleSoft Pension Administration function that determines how much service credit an employee has accrued.

Service Buy Back

The process by which an employee repays a pension plan in order to restore service credit that was forfeited when the employee withdrawal previous contributions. Typically, employees withdraw contributions upon termination and initiate service buy back processing upon rehire.

Service Purchase

The process by which an employee gets additional pension service credit for periods not normally considered eligible. The employee “purchases” this service by paying into the plan.

Service Schedule

A table showing how much service an employee earns based on the number of hours the employee worked during the year or month.

SetID

The label that identifies a TableSet.

Sex Code

Used to indicate gender.

Shape

For a transaction, the set of inputs and outputs for that transaction. For a class, the data members of that class.

Share

A share of a company's stock. Stock options give you the option to purchase a certain number of shares of company stock.

Share Price

The price per share of a company's stock. See, also, "stock price."

Shareholder

Owner of one or more shares of stock in a corporation. Also known as a stockholder or investor.

Shares Available to Issue

The total number of shares authorized, less shares granted, plus cancellations that revert to the Plan pool.

Shares Cancelled

This is usually triggered by a specific event, such as termination of employment in which the unvested shares as of the date of termination are no longer available for future vesting and exercise. These shares are therefore canceled from the option and can be returned to the plan, retired to treasury or allocated back to a group.

Shares Exercisable

The number of shares that are vested and available for exercise.

Shares Exercised

The number of shares purchased upon exercise of a stock option.

Shares Expired

Option shares that no longer are exercisable at the end of the option term. The length of the option term is defined in option agreement. This date is usually the earlier of the exercise period for vested shares after termination of employment or the full length of the option term.

Shares Outstanding

The number of company shares currently held by shareholders, as tracked by the transfer agent

Shift [Time and Labor]

The block of hours that an employee works in a day, such as nine to five, four to eleven, or ten to six. In PeopleSoft Time and Labor, Shift is used as a template of clock hours for scheduling an employee or group of employees to be at work or available to work (on call). Shifts may be constant, rotating, repeating, and/or split; any given shift may or may not have an associated Shift Differential or Bonus. A shift is always associated with a Work Schedule, and consists of clock hour Start and Stop times (two to allow for split shifts), meal periods (two) and relief periods (two).

In PeopleSoft Workforce Analytics, the block of hours that an employee works in a day such as nine to five, four to eleven, or ten to six.

Shift Bonus

A fixed amount (either a flat dollar figure or stated in terms of an employee's rate) paid for working a particular Shift.

Shift Code

A numerical shift identifier that is unique within a SetID.

Shift Differential

Additional compensation paid an employee for time worked during certain shifts. Typically, shift differential is administered as a flat amount per shift, hour worked, and/or as a percentage of the amount paid per shift hour or shift worked.

In PeopleSoft Workforce Analytics, a premium paid for work over regular pay, for which employees on certain shifts may be eligible, such as double-time for late night shifts. Shift differentials are usually stated as an additional rate or factor.

Shift Elements

Individual components of a shift such as TRC start and stop time, duration (see Scheduling)

Shift Name

Customer defined nomenclature for a shift (see Scheduling)

Shift Type

A customer-defined classification associated with a shift. The shift type can be used in the evaluation of rules or exceptions (i.e. On Call) (see Scheduling)

Shift Type [Time and Labor]

Time and Labor defined classification of shifts. Valid shift categories include Flex, General and Elapsed. Shift categories are used in the creation of time reporter schedules (see Scheduling)

Shipping Priority Code

Shipping Priority Codes act as tie breakers during order reservation in PeopleSoft Inventory when different orders are scheduled for the same delivery date and time. When the reservation process in Inventory encounters a situation where there are more orders than available stock, the system reserves the order with the highest priority. If schedules are encountered with the same priority, orders are then considered by order date.

Short Sale

The sale of a security that is not owned or is not delivered at the time of the trade, necessitating its purchase or delivery some time in the future to "cover" the sale. A short sale is usually made with the expectation that the stock value will decline, so that the short seller can eventually cover at a price lower than the original sale, thus realizing a profit. At the time of the short sale, the broker borrows stock to deliver on the settlement date. A short sale can be "naked," in which case the seller does not deliver the shares being sold short and must provide the broker with collateral. Or the short sale can be "against the box," in which case the seller delivers the shares being sold short for the broker to hold "in the box" until the seller chooses to close out the short position.

Short-Swing Transaction

A purchase and sale, or sale and purchase, of the issuer's equity securities by an insider within a period of less than six months. See "Section 16(b)" above.

Short-term Variable Compensation

In PeopleSoft Workforce Analytics, this is cash compensation paid to a worker that is not fixed; includes bonuses and commissions.

Sibling

A tree node at the same level as another node, where both roll up into the same parent. A node can be a sibling, parent, and child all at the same time, depending on its location in the tree.

Sibling

A tree node at the same level as another node, where both roll up into the same parent. A node can be a sibling, parent, and child all at the same time, depending on its location in the tree.

Sibling

A tree node at the same level as another node, where both roll up into the same parent. A node can be a sibling, parent, and child all at the same time, depending on its location in the tree.

Sick Leave

Sick leave is accrued by full-time permanent/seasonal employees at the rate of 4 hours every biweekly pay period; for part-time permanent/seasonal employees, it is accrued at one hour for every 20 hours worked.

Simulated Workforce

In the PeopleSoft Workforce Rewards product, Manage Compensation module, the calculated Simulated Workforce = Existing Employees + New Simulated Employees + Reduced-Employees.

Single Life Annuity

A benefit payable during the lifetime of the participant, with no payments made after the death of the participant. Also referred to as a "life only annuity" or a "straight life annuity."

Single Signon

This refers to the process by which a user can, after being authenticated by one PeopleSoft application server, access a second PeopleSoft application server without entering a user ID or password.

SIREN Code (Système Informatique pour le Répertoire des Entreprises)

This stands for the Electronic List of Enterprises. The SIREN code is assigned to a company when it registers as a business with the French government, and identifies the purpose of the establishment for regulatory reporting purposes in France.

SIRET (Système Informatique pour le Répertoire des Établissements)

This stands for Electronic List of Entities. In France the SIRET is an identifying number given to a French business by the INSEE, an official statistics and economics organization in France. The SIRET number is a combination of the SIREN and NIC numbers. This number is used by the tax and social security authorities to identify a business enterprise and its entities.

Site Tree

In PeopleSoft eStore, a hierarchical structure that controls navigation, as well as content and behavior within the header, footer, and left margin areas of the web page template.

Slice

The span of time into which an element is segmented as a result of element segmentation. Unlike a segment (or period), a slice does not represent a separate gross-to-net process since it affects only a limited set of elements within a period or segment. Like segments, slices have their own begin and end dates.

Slice Dimension

A model dimension used to restrict user access to the system. For example, a product manager's access to the system can be restricted to only the products he or she is responsible for by defining "Products" as a slice dimension, and assigning this person the members of the "Products" dimension he or she can access.

Slice and Dice

Another term for multidimensional analysis. When your data has three (or more) dimensions, you can think of it as being arranged in a cube (or hypercube), with each side representing a dimension. When you analyze the data, you "slice" off part of the cube or "dice" it to get to an individual cell.

Slotting

In PeopleSoft Workforce Rewards, a process by which the system establishes the target market compensation rates to use for compensating workers in non-benchmark jobs.

Social Security Number

Nine numeric digits assigned to an individual by the Social Security Administration. Also known as a Taxpayer Identification Number (TIN).

Source

The Source table stores valid journal entry and posting sources. These can include job titles (such as CFO), user IDs (such as CLERK123), PeopleSoft General Ledger processes (such as Consolidations), or other applications (such as PeopleSoft Payables).

Source Transaction

In commitment control, any transaction generated in a PeopleSoft or third-party application that is integrated with commitment control, and which can be checked against commitment control budgets. For example, a pre-encumbrance, encumbrance, expenditure, recognized revenue or collected revenue transaction.

Sparsity/Density

A multi-dimensional concept of whether data exists at intersections of dimensions. If a cube has many dimensions, but little or no data in some of those dimensions, the cube is considered sparse. Sparse cubes take up unnecessary disk space and reduce calculation performance. The goal is to create dense cubes and only use dimension intersections where data actually exists.

Special Accumulator

A device that accumulates earnings from different sources for a specific purpose. 401(k), pension and retirement plans use special accumulators. A 401(k) plan might use a special accumulator to calculate a deduction using regular, vacation, and overtime earnings. Special accumulators can add to or subtract from a pool of earnings.

Special Payments

A payment that occurs once or under special circumstances (e.g., back pay interest, lump sum leave, bond refund, longevity bonus, compensatory time reimbursement, death payment, severance pay, separation bonus, etc.).

Special Rates

Higher salary rates for specific grade levels and occupational groups determined by OPM for employees working in specific geographic areas. Each area is assigned a separate Schedule Number.

Specialist

A member of a stock exchange who maintains a fair and orderly market in one or more securities. A specialist or specialist unit performs two main functions. executing limit orders on behalf of other exchange members for a portion of the floor broker's commission, and buying or selling for the specialist's own account to counteract temporary imbalances in supply and demand, preventing wide swings in stock prices.

Specialized Experience

Experience that has equipped the applicant with the particular knowledge, skills, and abilities to perform successfully the duties of the position and is typically in or related to the work of the position to be filled.

SpeedChart

A user-defined shorthand key designating several ChartKeys to be used for voucher entry. Percentages can optionally be related to each ChartKey in a speedchart definition.

SpeedType

A code representing a combination of ChartField values. SpeedTypes simplify the entry of ChartFields commonly used together.

Spending Limits

See Planning Targets.

Split and Join

In PeopleSoft Demand Planning,the process of subdividing a forecast so that multiple users can make changes to their portions of the forecast. After changes are complete, the portions are joined back into a single forecast.

Split Deduction

Deduction that you create by splitting an existing deduction into two deduction items. The new deduction retains the original item ID with an added suffix number.

Split Shift

Periods of productive time split up by period of non-working time; example. a time reporter comes to work as a busboy for the lunch shift from 12-2 p.m. and then returns to work from 6-8 for the dinner shift (see Scheduling*)*

Split Stream Processing

The matching of a payment's cash information with the payment advice information when they have been received as separate transmissions through EDI and lockbox.

The uniting of the payment cash with the payment advice when they have been received as separate information through EDI.

Spokesmen Committee (Sprecherausschusse)

In Germany the Spokesmen Committee represents the interest of the management in your company before the ownership. The Spokesmen's Committee is consultative in nature, although they play a co-determination role on individual employment contracts, hiring, and dismissals. They also play a role in monitoring employment fairness, equity, and non-discrimination in terms of nationality, race, religion, sex, and age.

Spouse Demonstration J&S

In the PeopleSoft Pension Administration system, an informational-only form of pension payment that tells what the spouse's total benefit would have been if the retiree had chosen the spouse as the beneficiary rather than a nonspouse beneficiary. You cannot pay pension benefits based on this form because it is informational only.

Spouse Eligibility Alias

In PeopleSoft Pension Administration, a Custom Statement that defines any criteria that must be met before the plan will provide an Automatic Spouse Benefit. For example, the plan may require that the employee and spouse be married a full year before they are eligible for an automatic spouse benefit.

Spouse Eligibility Statement

See Spouse Eligibility Alias.

Spread

Depending on the context, either (1) the difference between the bid and asked prices for an over-the-counter stock, or (2) the difference between an option's exercise price and the market price at the time of exercise (i.e., the profit component of the exercise).

SQL Objects

Used to create rules that are more complicated than templates or actions and conditions allow—select statements, insert statements, table joins, and sub-queries

ST (Strategic Trust)

See Strategic Thrust

Staged Date

The date an item was received into the inventory business unit.

Staging ID

An identifier for a putaway plan. The inventory system sequentially assigns Staging IDs when it creates the putaway plan.

Standard Cost

A predetermined, fixed cost associated with an **Inventory Item** or **Forecast Item**, representing detailed estimates of each element of cost entering into the purchasing or manufacturing of an item. Standard cost is used when minor variations in an item's cost are not needed. The use of standard costs also enables management to determine how much an item should cost (Standard), look at how much it does cost (Actual), analyze the differences between the two and their causes (Variances), and compute economic order quantity.

Standard Form (SF)

A standardized form for interagency use by the Federal government. The SF prefix is the most common but not exclusive one in usage.

Standard Price

In PeopleSoft Demand Planning, the standard selling price associated with a **Forecast Item**. The price can be introduced into the system directly in forecast item maintenance or indirectly using the demand transfer interface. At higher levels in the view where there is no standard price available, the summarization function can be set up to develop one.

Standard Unit of Measure

The smallest unit of an item that a PeopleSoft application tracks.

Startup Data

In PeopleSoft Pension Administration, accrued Service, Cash Balance Account, or Employee Account data loaded into the system in the form of an opening balance and "as of" date. The alternative would be to load the entire accrual history.

State Record

The State Record is a PeopleSoft record, keyed by process instance, that must be created and maintained for each Application Engine program. The State Record defines the fields that an Application Engine program uses to pass values from one SQL statement to another.

Static Group

An employee group in Time and Labor that enables you to control its creation and maintenance. The group remains the same at all times until you change it.

Static Policy Controls

Determines how a static (versus time-phased) **Inventory Policy** is to be calculated. Static controls use period and average methods and their arguments.

Statistical Account

An account that has an associated unit of measure, used for tracking and monitoring statistical data. For example, the Workstations account uses EA (each) as a generic unit of measure, while the Floor Space statistical account might use square feet and the Work Days account would use days.

Statistical Code

The unit of measure used for tracking and monitoring statistical data. For example, using a statistical code of WS may represent the number of Workstations.

Statistical Forecast

In PeopleSoft Demand Planning,, a forecast developed at each level of the forecast pyramid and that considers the item's history in isolation.

Status Checking

In PeopleSoft Projects, a control feature that can be applied to transactions coming into Projects from cost feeder systems. If the incoming transaction does not conform to predetermined status and analysis conditions, an online warning will display or the transaction will be rejected.

Status Position Code

A code that identifies the various conditions of a position, e.g., frozen, classified, etc.

Statutory Account

Account required by a regulatory authority for recording and reporting financial results. In PeopleSoft, this is equivalent to the Alternate Account (ALTACCT) ChartField.

Step

A secondary level or subcategory within the primary pay level (depending upon pay plan, different employees may have a different number of steps within their primary pay level).

Step Progression

In PeopleSoft Workforce Analytics, a pay increase granted to an employee or group whose salary plan includes steps within grades. Each step increase is a step up the pay range for the employee.

Stock

In corporate finance, the form in which an owner's interest is represented, distributed in units known as shares.

Stock Administrator

An individual who administers and manages the corporation's benefits and/or equity compensation plans. This individual serves as the contact for transfer agent and broker inquiries. Stock Administrators manage Stock Option Plans, Employee Stock Purchase Plans, Restricted Stock Award Plans, and Stock Bonus Plans.

Stock Appreciation Rights (SAR)

A contractual right to receive, either in cash or employer stock, the appreciation in the value of the employer's stock over a certain period of time. A SAR can be used alone or in tandem with Incentive Stock Options (ISO/SAR) or Nonqualified Stock Options (NQ/SAR). PeopleSoft Stock Administration supports only tandem SAR's.

Stock Awards

Stock allocations that are processed in the Manage Variable Compensation business process. Stock Administration creates stock grants from finalized stock awards.

Stock Exchange

An organized marketplace in which bonds, stocks, and common stock equivalents are traded by members of the exchange, acting as agents (brokers) and as principals (dealers or traders). Such exchanges have a physical location where brokers and dealers meet to execute orders to buy and sell securities. Each exchange sets its own requirements for membership.

Stock Option

A contractual right granted by the company, generally under a stock option plan, to purchase a specified number of shares of the company's stock at a specified price (the exercise price) for a specified period of time (generally five or ten years). Assuming that the exercise price is the

same as the fair market value on the grant date, the option will become more valuable if the fair market value goes up, because the option effectively gives the optionee the right to buy stock in the future at a discount.

Stock Price

The price per share of a company's stock. See, also, "share price.

Stock Purchase Participant

An individual who participates in the corporation's Stock Purchase Plan.

Stock Purchase Plan

A type of broad-based stock plan that permits participants to use payroll deductions accumulated over a period of time to acquire stock from the company.

Stock Split

A change in the capitalization of an issuer that increases or decreases the number of securities outstanding, and adjusts the value of the securities accordingly, without a corresponding change in the assets or capital of the issuer. For example, if an employee has options to purchase 25 shares at $10 per share and the company has a 2-for-1 stock split, the employee thereafter has the option to purchase 50 shares at $5 per share.

Stock Swaps

A payment method that can be used to cover the cost of the exercise price and taxes depending on whether it is allowed by the plan. When an employee elects to exercise a stock option by means of a stock swap, they surrender already-owned shares of stock to pay the total required option exercise price and/or taxes for the option being purchased. The surrendered shares are usually valued at the fair market value of the company's stock on the date of exercise.

Stock Trading Symbol

The three or four letter symbol used to identify a company's stock on the stock exchange where it trades. Also known as a "ticker symbol".

Stock Withholding

A cashless method of satisfying the withholding taxes due upon the exercise of a stock option by authorizing the company to withhold from the shares being exercised a number of shares equal to the taxes.

Stockholder of Record

Person or entity, often a broker or the Depository Trust Company, named on the issuer's or transfer agent's stock record books as the owner of shares held in "street name." The stockholder of record acts in part as a way of safekeeping stock certificates that might otherwise be lost by the beneficial owner, and also in order to keep the identity of the beneficial owner confidential from the company.

Stock-In Probability

A replenishment option for defining transfer parameters for PeopleSoft Demand Planning or Inventory Planning upload files. The option is the percentage of time you want to have the item on hand for the **Business Unit** and is used to calculate safety stock.

Stop Time

Out punch

Storage Area

A division of a **Business Unit** used to store material and to track **Inventory Transaction**. Storage areas might include shipping and receiving docks, staging areas, warehouse zones, and inspection and quality control departments. Each storage area can be divided into a maximum of four levels, with each level representing a physical subdivision of the area.

Storage Level

A hierarchical subdivision of a storage area.

Storage Location

The combination of a storage area and that area's most detailed storage level. This is the smallest definable physical space within an **Inventory Business Unit**.

Strategic Initiatives

In PeopleSoft Balanced Scorecard, actions the organization must take to implement strategy. May be temporary or short-term in nature.

Strategic Thrust (ST)

In PeopleSoft Balanced Scorecard, four to five statements or paragraphs that summarize the core components of an organization’s strategy. Strategic thrusts describe the key areas across which a scorecard is balanced. They are themes or goals your organization is striving to achieve; more specific descriptions of what you must do to achieve that goal are defined by critical success factors. Key performance indicators may be attached to strategic thrusts as long as there aren’t critical success factors below them, but typically strategic thrusts aren’t directly associated with key performance indicators.

Strategy Tree

In PeopleSoft Balanced Scorecard, the hierarchical relationships of the objectives your organization is striving to achieve. Used as the foundation for a scorecard, and typically balanced across four major categories: Financial, Customer, Learning and Growth, and Internal Processes. These are made up of Vision, Strategic Thrusts, and Critical Success Factors.

Stratification Engine

A support module that structures the volume of financial accounts and balances at a large financial institution to a manageable scale for processing by the PeopleSoft Funds Transfer Pricing (FTP) and PeopleSoft Risk Weighted Capital (RWC) applications. It categorizes data by a range of values and summarizes data based on rules you define for FTP and RWC.

Stratification Wizard

Stratification Wizard is a tool you can use to quickly create new stratification rules or update the existing rules. Stratification Wizard prompts you for each of the possible source and destination fields, grouping operations, and summarization actions to be performed. Stratification Wizard enables you to stratify your data according to tiers, discrete values, periodic increments, and numeric increments. It also enables you to leave the data aggregated.

Streams

An optional feature that enables you to reduce processing time by processing groups of payees simultaneously.

Street Name

See "Stockholder of Record".

Stretch

In a Goals Matrix performance scale (In PeopleSoft Workforce Analytics), this is the level of performance for which an employee achieves maximum pay out. Performance above this level receives no greater pay out.

Strike Price

The price per share which must be paid in order to exercise the stock option. The strike price is typically the fair market value of the stock on the grant date. Also known as the "exercise" or "grant" price.

String constant

String constants are delimited in PeopleCode by using either single (‘) or double (") quote marks.

Strip Funding

One of several methodologies used by PeopleSoft Funds Transfer Pricing (FTP) to derive maturity when calculating FTP rates based on matched maturity funding. This approach matches the projected cash for the instrument in each time period, with a specific cost of funds rate for that cash flow. The FTP rate for the instrument is then calculated by weighting the cost of funds rate for the cash flow in each time period by the term of the cash flow.

Structured Query Report (SQR)

A type of printed or displayed report generated from data extracted from a PeopleSoft SQL-based relational database. PeopleSoft applications provide a variety of standard SQRs that summarize table information and data. You can use these reports as is, customize them, or create your own.

Style File (Verity)

Collection style refers to a set of configuration options that are used to create the indexes associated with a collection. A collection has one collection style and it is defined in a set of style files before creating the collection.

SubCustomer Qualifier

A value that groups customers into a division for which you can generate detailed history, aging, events, and profiles.

Sub-Process Section

A type of section you can add to a process list. Sub-process sections are especially useful for performing iterative processes such as gross ups (calculating the gross amount for a given net amount). You can include conditional logic within a sub-process section.

Subscription

The process of mapping fields, selecting data parameters and submitting the information to an outside vendor.

Substitute Item

In PeopleSoft Manufacturing, an item that can be used when there are no primary components available in inventory or when there is a long-term shortage of the original item. The substitute item can be defined at three levels: setID, business unit/item, and bill of material/engineering bill of material.

Subtask

A lower-level Planning task in a schedule's hierarchy that rolls up into a parent task. For example, an operation performed on a production ID would be a subtask of the production order.

Summarization Process

See Rollup.

Summary ChartField

A feature for creating summary ledgers that roll up detail amounts based on specific detail values or on selected tree nodes. When detail values are summarized using tree nodes, summary ChartFields must be used in the summary ledger data record to accommodate the maximum length of a node name (20 characters).

Summary Forecast

In PeopleSoft Demand Planning, a type of forecast that results from adding up the adjusted forecast totals from the next lower level, meaning the sum of the children's forecasts for the parent. The summary forecast at level one (1) is always zero since there is not a logical lower level.

Summary Ledger

An accounting feature used primarily in allocations, inquiries, and PS/nVision reporting to store combined account balances from detail ledgers. They increase the speed and efficiency of reporting by eliminating the need to summarize detail ledger balances each time a report is requested. Instead, detail balances are summarized in a background process according to user-specified criteria and stored on summary ledgers. The summary ledgers are then accessed directly for reporting.

Summary Tree

A tree used to roll up accounts for each type of report in summary ledgers. In effect, summary trees enable you to define "trees on trees." In a summary tree, the detail values are really nodes on a detail tree or another summary tree (known as the *basis* tree). A summary tree structure specifies the details on which the summary trees are to be built.

Summed Adjustment Type

When the system finds multiple summed discounts, they are added together, and applied once.

Super Tree Utility

A PeopleSoft Enterprise Warehouse utility that enables you to combine multiple effective dates of a tree into one. This super tree contains all tree changes for a certain period of time enabling you to analyze effective dated trees more easily.

Supplemental IRR

An IRR type used when a federal employee's retirement deductions were originally under-reported. An agency can create a Supplemental IRR to report the difference in the retirement deduction amount to the Office of Personnel Management (OPM).

Supplemental Tax Method

A payroll tax calculation method that uses a straight percent rather than allowances. The percentage depends on state requirements.

Supply Chain Warehouse

See Warehouses.

Support Costs

Activity costs not directly connected to production.

Support Modules

The support modules are a collection of engines and analysis models that derive values, rates, financial calculations, and prices. PeopleSoft Analytic Applications take this data and use it for further processing. The support modules perform processes that are used in the financial services industry. They are only utilized by two of the PeopleSoft Analytic Applications: Risk Weighted Capital (RWC) and Funds Transfer Pricing (FTP).

Support Team

A group of people working together to sell to and/or support a customer. You can assemble support teams and associate them with ship to customers, quotes, sales orders, and **Buying Agreement**.

Supporting Element Overrides

Provide a mechanism to override various supporting element types, such as brackets, dates, durations, formulas, and variables, at various different levels.

Supporting Elements

Supporting elements are building blocks for other elements. In PeopleSoft Global Payroll, they are used in combination with other elements to create rules. They are not stand-alone. Typical supporting elements are arrays, brackets, rounding rules, and fictitious calculations.

Suspend Exercise

As a condition of a leave of absence, a company may stipulate to restrict the exercise of shares during the leave or for a period of time. Only applicable if the Stock Action is LOA.

Suspend Vesting

As a condition of a leave of absence, a company may stipulate to suspend vesting of shares. Only applicable if the Stock Action is LOA. If a company does not Suspend Vesting then the Vest Deferral Grace Period and Service Rule are not applicable.

Suspended Item

In PeopleSoft Demand Planning, an item suspended by the system. The suspension is due to the lack of demand for the number of periods defined for the item's **Control Group**.

System Element

In PeopleSoft Global Payroll, system elements are delivered and maintained by Peoplesoft. There are two types of system elements: database system elements and system-computed elements. Database system elements contain payee-related data that can be used frequently in a calculation, such as department ID, location, and personal data. System-computed elements are automatically populated by the payroll process.

System Functions

A list of all activities that the system supports, along with their associated General Ledger distributions.

System-Defined Count

The PeopleSoft Inventory feature that employs user-defined criteria to begin the count creation process.

System-Defined History

Any statistical information updated by the posting and aging programs, maintained to reflect customer credit standing.

T

T+3

The obligation in the brokerage business to settle securities trades by the third day following the trade date. "Settlement" occurs when the seller receives the sales price (less the broker's commission) and the buyer receives the shares.

Table

The underlying format in which data is stored by columns (fields) and rows (records, or instances).

Table Lookup

A utility in PeopleSoft Pension Administration that finds an unknown value based on a known one in your data set. For example, you can look up an interest rate based on a year, or an actuarial factor based on an employee's age.

TableMaps

In the PeopleSoft Enterprise Warehouse, TableMaps define the physical relationships between related tables. TableMaps allow you to define "families" of related tables and the columns that define the key relationships between the tables.

TableSet

A group of rows across control tables identified by the same SetID.

TableSet Sharing

Specifying the control table data for each business unit so that redundancy is eliminated.

Tardy

The circumstance when a time reporter reports for work after the scheduled start time

Target

In a Goals Matrix performance scale (In PeopleSoft Workforce Analytics), this is the performance level your organization establishes as the norm for performance and pay out.

Target Cost

A desired target cost (for production, engineering or marketing) is found by subtracting the desired profit margin from a competitive or estimated price.

Target Currency

The value of the entry currency or currencies converted to a single currency for budget viewing and inquiry purposes.

Target End Date

The intended end date for employee schedules in Time and Labor reporting. You establish a target end date, and depending on the work schedule templates, the application determines what the actual end date should be.

Target Grade

Highest obtainable grade for a position.

Target Matrix

In PeopleSoft Workforce Rewards, the Target Matrix defines the level of award to be paid based upon a predetermined level of performance that a defined measure is evaluated against.

Target Plan

In PeopleSoft Workforce Rewards, a plan for distributing compensation awards, in which the level of the award is linked directly to a predetermined level of performance that a defined measure is evaluated against. For example, a Target Plan for a Sales group might be as follows: "As a group, increase the business unit's net income by 10% and each member of the group will be eligible for an award equal to 5% of base pay."

Target Rate

In PeopleSoft Workforce Rewards, calculated market rates you choose to use as new target pay rates for the jobs in your organization. Think of these target rates as pay guidelines, in support of your company's overall pay strategy.

Target View

In PeopleSoft Demand Planning, the of the views being reconciled during the **Cross-View Reconciliation** process. During the process, fields for reconciliation are defined for both a source and target view.

Tariff

In Germany a Tariff is a contract between the employee's unions and the employers' association, defining labor agreements on issues such as standard working hours, income, and vacation. This contract is valid for all of Germany for the business or industrial sector the company is working in (such as ***Banking*** or ***Metal***).

Tariff Area

In Germany, additional labor agreement terms beyond those in the Tariff, such as salary plans or employee reviews, can also be applied based upon the Tariff Area. The Tariff Area is often split along regional lines (such as ***Bavaria*** or ***Berlin***).

Task

See **Manufacturing Task** and **Planning Task**

Task

A piece of work assigned to or demanded of a person; a unit of work (see Time Reporting)

Task Entity

Individual component of a task; for example, Project ID, Activity ID, Work order, Department, Company, Business Unit (see Time Reporting).

Task Profile

A way of viewing or establishing where to allocate employee task information for a day and time. The task profile fields that appear on the page are established by the *Task Profile Template.*

Task Profile

Entity that establishes the default values for optional and required task elements. This can be for single or multiple tasks. (i.e. default values based on hours allocation, percentage distribution, equally distributed or by prompt) (see Time Reporting)

Task Rules

A methodology that is applied to scheduled, reported and payable time to allocate or redistribute task assignments (see Time Administration)

Task Transfer

Department transfer

Task Values

The customer defined value for a specific task element i.e. Customer 1, project 1, etc (see Time Reporting)

Taskgroup

Identifies the default time reporting templates, task template, and task profile(s) for time reporters that share the same task reporting requirements.

Taxable Benefits

Any employer contributions that are subject to Federal Withholding Tax.

TDS (Transfert de Données Sociales)

In France the TDS is a social security transfer report, submitted on magnetic media to the government.

Team Member

An individual who is part of a support team. Each team member may be in a commissionable or non-commissionable role.

Technical Scenario

In PeopleSoft Enterprise Warehouse technical scenarios allow you to set up the object type values that the Resolver uses to chunk the record/TableMap you'll resolve. Technical scenarios allow technical users to define chunking criteria that enable chunks to run in parallel. This allows for parallel data processing.

Template

A portal template is simply HTML code, associated with a web page, to define the style and layout of the page. Templates allow a developer to build an HTML page by combining HTML from a number of sources. Templates do two basic things: define the layout of the page, and define where to get HTML for each part of the page.

Template Pagelet

One piece of an overall template. For example, in a given template, there may be one template pagelet for the universal navigation header and one template pagelet for the target content.

Temporal Constraint

A relationship between Planning tasks that defines their sequence and timing in a schedule. Temporal constraints cannot be violated by the Optimizer. PeopleSoft Planning constraints include finish to start, start to start, finish to finish, start to finish.

Temporary Continuation of Coverage (TCC)

The TCC program, as prescribed by the OPM, requires Federal agencies to provide to separating Federal employees the opportunity to temporarily continue their FEHB coverage for up to 18 months (unless involuntarily separated because of gross misconduct), provided the individual pays the full cost of coverage, including both the employee and government share and a two percent administrative charge. Agencies may elect to provide this service in-house or enter into cross-servicing agreements with another Federal agency.

Tenor

Used by the PeopleSoft Funds Transfer Pricing (FTP) application to refer to the maturity of an instrument. It represents the length of time an instrument is available as either a source or use of funds. The FTP application calculates the transfer price for an instrument, based on the marginal cost of funds of similar liquidity and tenor.

Term Certain and Continuous Payment Option

See Certain and Continuous Payment Option.

Term Certain Payment Option

See Certain Only Payment Option.

Termination

A transaction in which an employee ceases to be an employee of the corporation.

Threshold

In a Goals Matrix performance scale (In PeopleSoft Workforce Analytics), this is the minimum threshold for adequate performance, the designated level of performance below which it is inappropriate to pay incentives.

Think-time process

Think-time functions suspend processing either until the user has taken some action (such as clicking a button in a message box), or until an external process has run to completion (for example, a remote process).

Three-Tier

A three-tier architecture introduces an intermediary application server between the client workstation and the database server to improve performance. Within PeopleSoft, the application server sends the SQL to the database and then returns results to the client in the form of lightweight Tuxedo messages.

Threshold Checking

In PeopleSoft Projects, a control feature that can be applied to transactions entered directly into Projects or integrated into Projects via the INTFC_PROJ_RES table. If the transaction exceeds a predefined tolerance, a warning will display or the transaction will be rejected.

Thrift Savings Plan (TSP)

A voluntary retirement savings and investment plan for Federal employees administered by the Federal Thrift Investment Board.

Ticker Symbol

The three or four letter symbol used to identify a company's stock on the stock exchange where it trades. Also known as a "stock trading symbol".

Tiers

In the financial services industry, Tiers are ranges that you set up for stratifying your instrument data (such as amounts, rates, and numbers) into specific groups. You define Tiers within Stratification Engine.

Time Administration

A process which provides four [separate] different online tools for creating, maintaining, and applying an organization's compensation, task, and exception rules to both reported and scheduled time. templates, actions and condition, SQL objects, and user exits

Time and Labor Period

A distinct, configurable period of time used by all the PeopleSoft Time and Labor processes (see Time Reporting)

Time and Labor User

Either a Time Reporter or a Time Manager

Time Capture Device

Third party system or methodology for collecting elapsed or time capture device time, i.e., time capture device, IVR, Fax, etc. (see Time Reporting)

Time Collection

A Time and Labor feature that collects positive and exception time reports, applies appropriate business rules and edits to the reported time to ensure validity and reasonableness, and returns errors and questionable items to the time reporter for correction or scrutiny. Time collection is also responsible for scrutinizing future (previously) posted time information for correctness when those reports are ready for use.

Time Collection Device

A group of time collection device lumped together and named for ease of assignment to employees. In other words. clock group 1 is made up of clock 1, and clock 2. Employee 123 is assigned to clock group 1 and can then punch in at either clock 1 or clock 2. (see Understanding Time Collection Device)

Time Collection Device time

Reporting time by recording actual starts and stop times (see Time Reporting)

Time Dimension

Determines how date-related information is presented in a **Cube View**. This dimension defaults to a two-level hierarchy consisting of the **Inventory Policy** year and a standard period, such as monthly.

Time Fence

In PeopleSoft Planning, a user-defined parameter that specifies the business rules to be used in the generation of the plan. PeopleSoft Planning time-fence types include start of time, end of time, planning close date (demand time fence), purchase order fence, leveling fence, action message cutoff, and planning time fence.

Time Manager

An individual who supervises Time Reporters

Time Period

A period of Time used in Time and Labor rules processing. You can categorize time periods in terms of days, weeks, or months. You establish day, week, or month-type periods for use when you apply rules for compensation, holidays, and so on.

Time Report

A payroll time and/or labor distribution time report for an employee for any date within the employee's current period.

Time Reporter

Any employee or contractor for who time is reported or generated in PeopleSoft Time and Labor.

Time Reporter Information

Values associated with the Time Reporter that are displayed when entering or viewing reported time and facilitate the processes of Time Reporting and Time Management (see Time Reporting)

Time Reporting

Any information required by a business unit that can be attributed to an individual employee (worker/contractor) and can be expressed in hours.

Time Reporting Code

A hybrid of two PeopleSoft objects: the Payroll Earnings Type and the Human Resources Absence Type. The Time Reporting Code represents the level at which a business actually needs to track employee time to support all of its administrative and compensation needs.

Time Reporting Code Type

Categorization of a time reporting code. Valid categories include. units, amounts, hours or a combination of hours and amounts (see Time Reporting)

Time Reporting Group

See Group [Time and Labor].

Time Segment

For Service, Cash Balance Accounts, and Employee Accounts, employees can accrue benefits differently at different times. The period of time during which employees use a particular rule is that rule's time segment.

TimeSpans

Relative periods, such as year-to-date or current period, that can be used in various PeopleSoft General Ledger functions and reports when a "rolling" time frame, rather than a specific date, is required. TimeSpans can also be used with Flexible Formulas in PeopleSoft Projects.

Tolerance

In PeopleSoft Projects, a value that is defined at either the project or activity level as either a percentage of a project's funding or an actual amount.

In commitment control, the percentage over budget that you allow, excluding revenues applied to increase budget limits, before the system creates an exception.

Total Authorized But Unissued

The combined total number of shares from Shares Available to Issue plus Total Options Outstanding.

Total Compensation

In PeopleSoft Workforce Analytics, this is generally the officially recognized compensation provided an employee in the course of their employment with an organization; includes both direct compensation and benefits compensation.

Total Compensation Management

The ability to track and report on all types of cash programs, non-cash programs, benefits and deferred compensation for all current workers, ex-workers and individuals associated with ex-workers who receive compensation due to the employment of the ex-worker.

Total Non-Compensation

In PeopleSoft Workforce Analytics, generally, this is the often unrecognized compensation an employee receives in the course of their employment with an organization; it includes Learning and Development compensation such as training, and Workplace Environment Compensation such as telecommuting privileges or other prerequisites.

Total Options Outstanding

The number of company shares currently held by shareholders as tracked by the transfer agent. Derived by using the number of Prior Outstanding plus Grants, less Exercises and less Cancellations.

Total Rewards

In PeopleSoft Workforce Analytics, this is the total rewards provided to an employee by their employment with an organization; it includes their officially recognized total compensation, and less often recognized total non-compensation.

Tour of Duty

The scheduled days and hours per day of attendance at a duty station for an employee.

Tracking Signals

PeopleSoft Demand Planning,a forecasting tool that detects bias in the forecast and provides an early warning of an unstable forecast. There are six tracking signals associated with each **Forecast Item** that correspond to the six most recent historical periods.

Trade Payment

An authorization for a customer deduction in a Promotion application.

Training Report 2483

The Training Report 2483 is a French regulatory report used to declare vocational training your company has provided to your employees. It is also known as the Declaration 2483 Report. The purpose of the report is to receive tax deductions from the government based upon the amount of money your company has spent on training.

Transaction

A named command with optional named and typed inputs and outputs. The associated external system or the Business Interlink Plug-in understands this command. The types of inputs and outputs are based on a set of generic types.

Transaction

A named command with optional named and typed inputs and outputs. The associated external system or the Business Interlink Plug-in understands this command. The types of inputs and outputs are based on a set of generic types.

See also **Inventory Transaction** or, for PeopleSoft Projects, **Resource Transaction.**

Transaction catalog

Lists transactions used to interface to the external system.

Transaction Code

In PeopleSoft Projects, an additional field on each resource transaction that is used in conjunction with accounting entry templates. Transaction codes enable you to deal with exceptions to your accounting entry templates without having to create additional transaction types. You can set up separate accounting entry templates for resource transactions containing the transaction codes you create. The accounting entry templates for those resource transactions lines can then use the same transaction types, but specify different accounts.

In PeopleSoft Asset Management, transaction codes identify special asset transactions and are used in conjunction with transaction type to create accounting entries.

Transaction Code

Identifies what action has taken place against the position.

Transaction Costing

See Multidimensional Costing

Transaction Currency

In the financial services industry, the original currency in which a company conducts its business activities. When a company has multinational operations, it may use different transaction currencies. These are translated to the base currency for consolidation and reporting of financial results.

Transaction Date

The date a transaction actually occurred as opposed to the date the transaction is recognized—the accounting date (although the two dates can be the same).

Transaction Dated

Data aggregated over a date range.

Transaction group

The package can contain one or more transaction groups. Each transaction group is a set of transactions of the same type, with the same trading partners involved.

Transaction Loader

The SQR in PeopleSoft Asset Management that transfers load lines from the Loader tables into the PeopleSoft Asset Management Tables as assets and open transactions.

Transaction Tables

In the PeopleSoft Enterprise Warehouse, these are tables that contain dynamic information and are keyed by business units.

Transaction Type

The building blocks of accounting entry templates in PeopleSoft Asset Management and Projects. For each transaction type you create you define specific transaction lines. The transaction lines are then transferred into accounting entry templates. In the accounting entry templates each transaction line is assigned a specific general ledger account.

Transactional System

A business application for performing the business transactions that keep your company running. Transactional applications, and the databases that support them, are optimized for quick transaction processing. Because they are constantly changing and are not optimized for data retrieval, transactional system databases are not usually the best source of data for analysis.

Transfer Agent

An individual or firm who that keeps a record of your shareholders and the number of shares they own. Transfer Agents also issue new share certificates and cancel old certificates. Unlike Brokers, Transfer Agents are not responsible for selling stocks. Instead they are primarily concerned with maintaining records on all stocks which your company has issued.

Transfer Forecast

In PeopleSoft Inventory Planning, a Generation process option that transfers the forecast from the target view in Demand Planning forecasts. The process only transfers items from Demand Planning that have been set to update the **Inventory Policy**.

Transfer Punch

The start of a work period that specifically denotes a change in task and usually compensation-related characteristics

Transfer Type

An interunit transfer setting PeopleSoft Production Planning and Enterprise Planning use to determine where it will obtain item data for transfer tasks. If the type is a supply or demand transfer task, the Planning engine only processes the transfer item for a single location, reducing the time for plan processing. If the transfer type value is both, the Planning engine processes the transfer item using data from both the To and From units.

Transfer Worksheet

A work space for transferring an open item from one customer to another.

Transferable Stock Options

Options that may be transferred by the optionee, generally only to a family member or to a trust, limited partnership or other entity for the benefit of family members, or to a charity.

Translate Table

A system edit table that stores codes and translate values for the miscellaneous fields on the database that do not warrant individual edit tables of their own.

Translate Table

A system edit table that stores codes and translate values for the miscellaneous fields on the database that do not warrant individual edit tables of their own. In most cases PeopleSoft maintains the Translate Table.

Transport Rate

The Transport is a statutory deduction in France. Each establishment has a rate, and the URSSAF notifies establishments of this rate on a yearly basis. This deduction is used by the region to subsidize transportation, and maintain and build roads.

Transportation Lead Times

The transportation lead time is the in-transit interval from the date and time a shipment leaves your warehouse (**Inventory Business Unit**) to the date and time it arrives at your customer's receiving dock. The transportation lead time is used in calculating the scheduled shipment and scheduled arrival dates on the order when you enter either a requested arrival date or a requested shipment date.

Travel And Relocation Date

Length of time an employee must remain in the Government after the Government has paid to relocate him/her from one official duty station to another or for initial appointment.

TRC Program

A program that runs the level at which an organization actually needs to track employee time to support all of its administrative and compensation needs. TRCs are assigned to TRC Programs, which are ultimately assigned to workgroups. Multiple Workgroups can share these TRC Programs.

Treasury Interface files

These are DOS-based files generated by PeopleSoft in accordance with FMS file layouts for transmission of payment data to one of the FMS' Regional Financial Centers.

Treasury Position Code

In the financial services industry, this is a lookup code used for off-balance sheet treasury position accounts, such as foreign exchange, derivatives, precious metals, or any other account position that is the result of trading room and treasury operations.

Treasury Stock

Shares of a company's stock that have been repurchased or otherwise reacquired by the company and are "held in treasury." Whether the treasury shares count as "issued" or as "outstanding" shares of the company is a matter of state corporate law. Generally, a company may not vote its own shares held in treasury.

Treasury Stock Method

The method of calculating primary and fully diluted earnings per share when common stock equivalents such as unexercised stock options exist. Required under generally accepted accounting principles.

Tree

The graphical hierarchy in PeopleSoft systems that displays the relationship between all accounting units (for example, corporate divisions, projects, reporting groups, account numbers) and determines roll-up hierarchies.

Tree Compare Utility

A PeopleSoft Enterprise Warehouse utility that enables you to compare effective dates for trees. The results page shows nodes that have been added, deleted, or moved from one parent to another. You may also view the detail objects that have changed.

Tree control

Tree Control is a hierarchical search tool that you can imbed in a panel. Tree Controls give the user a view of hierarchical data structures and enable them to drill down through the hierarchy to a particular row of data.

Tree Denormalizer

The Tree Denormalizer Application Engine process converts trees into multi-column data format so they can be used by third-party OLAP or ROLAP tools.

Trigger

See Event Trigger.

Trustee Extract

A PeopleSoft Pension Administration data extract containing data that a third party needs in order to produce pension checks.

Turnover Costing

In PeopleSoft Workforce Analytics, this is a calculation of the cost to the organization of employee turnover, in dollars.

Turnover Rate

In PeopleSoft Workforce Analytics, the rate that employee's are leaving the company.

TUXEDO

BEA's middleware product used to manage transaction queues, server process initiation, system administration, time-outs, data encryption, compression, logging and other application server processing.

Two-Tier

A two-tier architecture refers to the traditional client/server model in which a client workstation connects to and sends SQL directly to the database server.

Type of Appointment

Indicates the specific type of appointment, e.g., part-time permanent, full time temporary, etc.

U

Underlying Security

The security underlying a stock option that an optionee has the right to buy, or the security underlying a convertible security.

Underpayment Adjustment Limit

The maximum amount or percent above which underpayment adjustments are not allowed for a given business unit.

Underwater Option

When the current market price is below the option exercise price. When an option is underwater, it would cost more than the underlying stock is worth to exercise the option. Such options are also described as being "out-of-the-money."

Underwriter

An investment banking firm that actually buys the shares from the company in a public offering and then resells them (at a slightly higher price) to its customers.

Unexpected Losses

In the financial services industry, these occur when the economic capital is exhausted and the insolvency rate is exceeded. Unexpected losses are determined by a targeted insolvency rate (confidence level); for example, a 99.7% confidence level indicates that there is a 0.03% estimated probability that the unexpected losses will exceed economic capital (or shareholder equity).

Union Code

Part of a group of defaults assigned to job codes. Union code may be used by human resources to group similar jobs or bargaining units together, dependent on individual company parameters.

Unit Code

In the financial services industry, Unit Code is used as an alternate means of measuring the relative size of companies participating in external surveys. A typical measure would be the number of employees in a company. The concept of unit is generic enough that the units can be other measures besides number of employees. For example, in the hospital industry the unit could be the number of hospital beds. Or in the hotel industry the unit could be the number of rooms.

Unit of Measure (UOM)

A type of unit used for quantifying in PeopleSoft systems. Depending on the application, units of measure might describe dimensions, weights, volumes, or amounts of locations, containers, or business activities. Examples include inches, pounds, workhours, and standard cost dollars.

Unit of work

Each transaction group includes one or more individual units of work. A unit of work is a single transaction that you want to commit or rollback as a whole.

Unitize Assets

The process of unitizing a single load line, usually originating from a different application, into multiple assets in PeopleSoft Asset Management.

Univariate Forecasting Technique

In Enterprise Planning and Simulation, the Univariate Forecasting Technique is a forecasting method that uses only the recorded history for the value to forecast its future.

Universal Navigation Header

Every PeopleSoft portal includes the universal navigation header, intended to appear at the top of every page as long as the user is signed on to the portal. In addition to providing access to the standard navigation buttons (like Home, Favorites, and signoff) the universal navigation header can also display a welcome message for each user.

Unscheduled Punch

A punch that is made be a time reporter who was not scheduled (see Time Reporting)

Unvested Shares

Unvested stock options are options that have not vested and, therefore, are not exercisable.

URI

A URI does not include the query string (the text following a ? on the URL). You can think of it as a subset of the URL that points to the resource, but does not include any parameters being passed to that resource. From the above example, the URI portion of the URL is as follows:

http://serverx/InternetClient/InternetClientServlet

URL

In this document, the term URL refers to the entire query string. The following is an example of a URL:

http://serverx/InternetClient/InternetClientServlet?ICType=Script&ICScriptProgramName=WEBLIB_BEN_401k.PAGES.FieldFormula.iScript_Home401k

URSSAF Code

The URSSAF is the body responsible for ensuring payment of Social Security contributions by all French employers.

Useful Life

The amount of time an asset may be depreciated.

User Data

PeopleSoft Demand Planning,data held in user-defined fields. These fields provide for storage of additional data that is not supplied by the standard set of fields in the system. The fields can also become part of the key for the **Forecast Item** at each level within the **Forecast View**.

User-Defined History

A summary of customer receivables activity that is defined by the user.

User-Field Code

PeopleSoft Demand Planning,a definition of a set of user-defined fields that contain data specific to the installation.

V

Valuation

The way a company represents the value of a non-monetary award such as stock.

Value allocation

A process in PeopleSoft Workforce Planning, by which you assign an overall monetary value to a competency strategy for your organization, and assign a weight or importance to the roles,

competencies and accomplishments in the strategy. The system then allocates a breakdown of the value to roles, competencies, and accomplishments in the strategy based on their relative weighting or importance.

Value Object

In the PeopleSoft Enterprise Warehouse, Value Objects are a metadata layer that provides descriptive information about fields and values. Value Objects are used as constants in Data Manager target object definition.

Variable

Temporary storage for use or defined information used in the creation and application of rules (see Time Administration)

Variable [Global Payroll]

An element type that defines and stores values such as a character, date, or number. You can use variables to create generic formulas for situations where you use the same values over and over again in a calculation.

Variable Compensation

In PeopleSoft Workforce Analytics, direct compensation that is not fixed, that is paid out in variable amounts, such as bonuses and commissions.

Variable Plan.

A plan in which either the number of shares and/or the price at which they will be issued is not known on the grant date.

VAT Account Type

A code that identifies the different types of accounting entries that must be created for VAT transactions. These codes are also used to categorize transactions in the VAT transaction table. The account type is used in conjunction with the VAT code and VAT transaction type to determine the VAT ChartFields used for a given VAT accounting entry.

VAT Apportionment

For mixed activity, VAT apportionment is the mechanism that allows you to specify the ratio of taxable activity to exempt activity for individual ChartFields.

VAT Calculation Method

Options are Net or Gross. When calculating VAT at net, the early payment discount is applied to the goods amount before calculating the VAT. The amount of VAT calculated using this method is the amount that is to be paid, regardless of whether the early payment discount is actually taken at time of payment. When calculating VAT at gross, the VAT is initially calculated based on the gross transaction amount. The early payment discount is not taken

into account at this point. However, in some countries an adjustment is made to the VAT amount at the time of payment, if the early payment discount is taken.

VAT Calculation Type

Options include Exclusive or Inclusive. If exclusive, the VAT amount is stated separately from the goods amount. If inclusive, the VAT is not stated separately but is included with the goods amount.

VAT Code

The tax code used to define a percentage the system uses to determine the VAT amount. The VAT code is similar to the sales and use tax code, with a few exceptions. The tax authority tied to the VAT code generally consists of a single authority, and the ChartFields for a VAT code don't reside with the tax authority but are determined by the combination of the VAT code, VAT account type, and VAT transaction type.

VAT Declaration Point

When VAT transaction information is declarable for reporting purposes. Options include Invoice or Payment. If you choose invoice, the system will recognize VAT at invoice time; if you choose payment, the system will recognize VAT at the time of payment.

VAT Entity

The level or entity within an organization at which VAT reporting is performed. VAT entities can be registered for VAT in multiple countries, but only one country can be designated as the VAT entity's home country. VAT and Intrastat reporting information and VAT default information are defined for each country in which the VAT entity is registered. You may also specify any VAT exceptions—either exoneration or suspension from paying VAT—for any country in which the entity is registered.

VAT Exempt Supply or Purchase

A transaction where the product or item is non-taxable or exempt from VAT. No VAT code is associated with the transaction. Although no tax is applied to the transaction, the transaction is still logged in the VAT transaction table.

VAT Exonerated

A transaction where the purchaser has been determined as not subject to VAT. For these cases, there may be an exoneration certificate number tied to the purchasing entity (either the customer or the VAT entity) as proof of exoneration. A zero-rated VAT code should be associated with transactions where exoneration applies. The transaction is still logged in the VAT transaction table, but no tax is applied.

VAT Rebate Percent

Within Canada, for Public Service Bodies, the percentage of VAT that is not normally recoverable but which may be refunded in the form of a tax rebate.

VAT Recoverability Percent

The percent of VAT that's recoverable.

VAT Registration Countries

Country codes associated with a VAT registration number for a particular customer or VAT entity.

VAT Transaction Table

Stores detailed transaction information for VAT reporting. It is the primary source of information for all VAT reports. Each application is responsible for writing to this table and also to a cross-reference table used to link entries in the VAT transaction table with entries within each application.

VAT Transaction Type

Used to categorize VAT transactions according to particular VAT accounting and reporting requirements. The VAT code and the VAT transaction type are used in conjunction with the VAT account type to obtain the ChartFields for accounting entries.

VAT Treatment

A description of how the transaction must be treated for VAT purposes. This is used to determine how VAT defaults are applied, what accounting entries are required, and how and if the transaction is reported on the VAT return.

VAT Use ID

A code used to identify the type of activity in which a purchased good or service will be used, and therefore to determine a recoverability percent and a rebate percent (when applicable) that will be applied to a transaction line. Activities are categorized as taxable, exempt, or mixed. Where activity is mixed, you may associate either the ratio of taxable activity to exempt activity directly with the Use ID, or you may indicate that this ratio is determined at the ChartField level.

VdkVgwKey

A key within a Verity BIF file for every document to be indexed. VdkVgwKey values must be unique across all collections that will be searched in any one application.

Vendor Draft

A draft issued by a vendor. PeopleSoft Receivables generates vendor drafts, provides a flexible worksheet environment for approval management, and enables discounted or standard submission for bank processing. PeopleSoft Payables receives vendor drafts and associates the appropriate vouchers.

Verity

The third-party search engine integrated with the PeopleSoft Portal.

Verity Fields

Verity fields are stored in the collection for retrieval and searching, and can be returned on a results list. Fields are defined in the BIF file and stored in the collection for retrieval and searching, and can be returned on a results list. Fields, like date and numeric fields can be used with the comparison operators (<,<=,>,>=).

Verity Thesaurus

The custom thesaurus consists of lists of synonyms defined in a synonym control file and can be used for synonym searching. After defining synonym lists in the control file, you use the mksyd utility to create a custom thesaurus (a control file which has the .syd extension) that the search engine uses.

Verity Topics

Verity applications can provide end users with predefined search criteria called *topics*. A topic is a named object that represents a concept, or subject area and can be used for synonym searching. It consists of words and phrases grouped together using the Verity query language in a tree-like structure. When provided, topics can be shared by all users.

Verity Zones

Zones are specific regions of a document to which searches can be limited. When the zone filter is used, the Verity engine builds zone information into the collection's full-word index. The index, enhanced with zone information, permits quick and efficient searches over zones. Searching a zone is faster than field searching. Zones are defined in the DAT file. The contents of a zone cannot be returned in the results list of an application.

Version

There can be up to five budget versions for each Budget Center level in a Budgeting Model. Budget versions are used to perform what-if analysis and comparisons of budget amounts before the user selects one version to submit as the Budget Center's budget plan. PeopleSoft Budgeting-specific.

Vest Deferral Grace Period

The specified period of time within which an optionee must return from leave to avoid having the vesting differed. Only applicable if the Stock Action is LOA. Suspend Vesting must be selected for this rule to be applicable.

Vest Deferral Grace Period Service Rule

If the company provides a vest deferral grace period, they may stipulate that only certain individuals are eligible for the grace period based on service with the company. Only applicable if the Stock Action is LOA. Suspend Vesting must be selected for this rule to be applicable.

Vest Immediately

A stock option plan may provide that upon specific types of terminations, or upon a case by case scenario, all unvested shares held by an individual can be made immediately vested as of a specific date. Some companies' plans provide that under certain circumstances, such as retirement, the vesting of option shares accelerates upon termination of employment. When this occurs, you must modify the vesting schedule before you terminate the individual.

Vested Shares

Option shares that are free of any ownership restriction. Generally, vested exercised shares are fully owned by the optionee, free from restrictions and freely tradable.

Vested Termination

The termination of an employee who has a vested benefit. The benefit is deferred until the participant reaches retirement age. The employee is considered “Terminated Vested,” “Term Vested,” or simply “TV.”

Vesting

The method by which a granted option becomes free of all restrictions and the Optionee has full rights to the shares.

Vesting Schedule (Template)

A convenient way to set up the framework for a vesting schedule that can be uniformly applied to individual options. When you grant stock options, you define a vesting schedule to determine the default-vesting schedule for the option.

Vesting Service

The service used to determine an employee’s vesting percentage. Rules for accruing vesting service may be different from rules for accruing other plan service credits.

VETS–100 Federal Contractor Report

This report is required of employers in the United States. It lists federal job classifications, and the number of employees and new hires in the last 12 months who are special disabled military veterans or Vietnam era military veterans. It also provides totals for each job classification of both veterans and non-veterans who hold these jobs.

View

PeopleSoft Demand Planning,a multilevel forecast structure. Each view is associated with a unique view ID and includes information that defines the view and structure type. The three types of views are working, disbursement, and dynamic.

For PeopleSoft Budgeting see Budget View.

Virtual Tasks

In Time and Labor, Virtual task data is associated with a taskgroup profile that defines common characteristics for a given Taskgroup and Task Profile ID. A single row of data is linked to multiple Earnings records for multiple employees. By minimizing the physical storage of daily task data we provide enhanced performance without limiting its functionality.

Vision

In PeopleSoft Balanced Scorecard, the overall mission of an organization. Usually the highest level on a strategy tree. Vision is optional; you aren't required to have a vision component on each strategy tree.

See also Strategy Tree

Volume

Total share volume traded in a stock during market hours.

W

WA (Workforce Analytics)

See PeopleSoft Workforce Analytics

Waiver Of An OPM Qualification Standard

Involves setting aside requirements in a published standard to place an employee in a particular position, usually to avoid some kind of hardship to the employee, such as in cases of RIF or administrative error on part of the agency. Extra training and/or skills development may be needed to help the employee adjust to the new position. Waivers are granted by OPM or an agency, as appropriate, on a case-by-case basis, and do not directly affect other positions in the organization.

Warehouses

A warehouse reporting and analysis solution that supports the specific PeopleSoft business application that warehouse is using. It consists of predefined ETL maps, data warehouse tools, and Data Mart definitions. The warehouses are we deliver are: PeopleSoft Financials Warehouse, PeopleSoft HRMS Warehouse, PeopleSoft CRM Warehouse, and PeopleSoft Supply Chain Warehouse.

Warning Exception

A transaction that exceeds the available funds but is allowed to continue to be posted against the budget. Warnings are informational only.

Warrant

A type of security, usually issued together with a bond or preferred stock, that entitles the holder to buy a proportionate amount of common stock at a specified price, usually higher than the market price at the time of issuance, for a period of years or to perpetuity. A warrant is usually issued as a sweetener, to enhance the marketability of the accompanying fixed income securities. Warrants are freely transferable and are traded on the major exchanges.

WCB

In Canadian provinces the Worker's Compensation Board (WCB) operates as an independent board, and thus would have different requirements in each province. For example, in British Columbia the organization is called the Worker's Compensation Board of British Columbia and in the Province of Quebec, the board is known as Commission de la Santé et de la Sécurité du Travail (CSST).

Weight

In PeopleSoft Planning, a user-defined value for the constraints that can be violated, determining how the schedules score will be calculated. Violations that are more critical to your schedule merit a higher weight.

Weight and Volume Pricing

You can price shipments by weight or volume to create price prices. Weight and Volume pricing requires using estimated shipments.

Weighted Average Cost of Funds

The projected principle payments for an instrument are used to derive a series of matched maturity funding rates, which in turn are used to calculate the overall base PeopleSoft Funds Transfer Pricing (FTP) rate. The Weighted Average Cost of Funds (WACF) method calculates a weighted average FTP rate where each of the funding rates is weighted by the principle payment amount and the term to maturity of the payment.

WFA (Workforce Analytics)

See PeopleSoft Workforce Analytics

WGI Due Date

Identifies the date of an employee's next within grade increase. Current policy is that the step increase is implemented on this date automatically unless prevented by the processing of an unsatisfactory performance appraisal.

WGI Non-Creditable Days

Total number of days that cause the WGI due date to be adjusted forward.

Whole Calendar Month

An instruction telling the system to use every day in each month for this time period. The system fills in the last day of the period according to the information you have entered.

Wildcard

You can replace the right-hand characters in a search field with a percent (%) wild card to query a range of values beginning with the remaining, left-hand characters. For example, by entering '2%' in a six-character field, you will receive a range of available values, such as 200000 through 299999 or 2aaaaa through 2zzzzz.

Window Period

The ten-day period, from the third to twelfth day after public release of a company's financial statement, when insiders may exercise their stock-appreciation rights without violating Securities and Exchange Commission rules for short-term trading.

Windows Client

Traditional PeopleSoft 32-bit client. Windows clients connect to the application server domain (Tuxedo) using a port number (or connection string) specified in PeopleSoft Configuration Manager.

WIP Replenishment Method

Designates how the PeopleSoft Flow Production request is communicated. For a replenishment method of Inventory, the Workflow, Pull Ticket, and Pull List replenishment methods are available. With feeder line replenishment, you can only use Pull Tickets.

WIP Replenishment Mode

Determines how PeopleSoft Flow Production is triggered to generate a replenishment request for an item. Replenishment options include Backflush, Manual, and Kanban Card.

WIP Replenishment Source

Determines where you send your PeopleSoft Flow Production replenishment request and what source supplies your WIP location. Options include Feeder, Inventory, and Vendor.

Withdrawal

An election not to continue participation in a stock purchase plan.

Withdrawal of Contributions

In a pension plan, the act of returning pension contributions, with interest, to an employee who is terminating. An employee who withdraws contributions typically forfeits all service associated with those contributions. If the employee is later rehired, repayment of contributions and interest typically reinstates the forfeited service.

Withholding

A deduction taken by employers out of taxable income of an individual. Typical withholding taxes include federal income taxes, federal social security, Medicare taxes, and state and local income taxes.

Within Grade Increase (WGI)

A longevity-based increase in salary based on predetermined time in grade requirements and acceptable performance.

Work Council (Comité d'Enterprise)

In France it is mandatory for companies with more than 50 employees to elect a Work Council to represent the employees in negotiations with management.

Work Effort

See Activity Type.

Worker

In PeopleSoft Workforce Analytics, workers are defined as anyone who performs functions for the organization, and receives compensation from the organization's operating expense funds in return. Workers can be direct employees or independent contractors. This includes individuals contracting business directly from the company or through an agency.

Work Group

In PeopleSoft Enterprise Performance Management, the work group is a grouping of employees that share a similar activity profile.

Work Period

A Days On/Days Off template; the smallest unit of time that a business uses to communicate with their employees regarding when to be and/or not to be at work (that is, time working and time not working). The work period can be any number of hours. Until clock hour reporting is implemented, the application does not care about the number of hours. The initial Time and Labor product will apply the work period to a calendar day.

Work Queue

In PeopleSoft Demand Planning and Inventory Planning, a feature for reviewing and working with exceptions created during the processing of forecasting and inventory data.

Work Schedule

A template consisting of a sequence of work periods (days) on and off, and the number of scheduled hours per work period. Work Schedules and Work Periods should not be confused with calendar days.

Worksheet

A way of presenting data to the user through a BAM interface that enables users to do in-depth analysis using pivoting tables, charts, notes, and history information.

Work Templates

Work templates describe your employee's work patterns. Work templates could apply to individuals or entire organizations. For instance, 9 AM to 5 PM, Monday through Friday is a fairly standard working week in organizations.

Workday

A 24-hour period rounded by daybreaker with one or more associated shifts (see Scheduling)

Workday Override

A function that allows a Time Manager to override a Time Reporter's schedule for a single workday. For example, Jane's long-term schedule assignment is Monday – Friday, 8.00 to 17.00. Due to an increase in production demand, her manager needs to schedule her to work 7.00 to 18.00 on Thursday, 16 March 2000. Her manager needs to be able to make this change to her schedule in the PeopleSoft Time and Labor system, so when Jane checks her schedule for this week, she'll see the revised schedule.

Worker

Workers can be defined as anyone who performs functions for the organization and receives compensation from the organization's operating expense funds in return. Workers can be direct employees or independent contractors (includes individuals contracting business directly from the company or through an agency).

Workers Compensation

The days an employee is on LWOP due to sustaining an injury or illness while on the job.

Workflow

The background process that creates a list of administrative actions based on your selection criteria and specifies the procedure associated with each action.

Workflow

The background process that creates a list of administrative actions based on your selection criteria and specifies the procedure associated with each action.

Workforce Monthly Report (Déclaration Mensuelle Obligatoire des Mouvements de Main D'oeuvre)

In France, companies that employ 50 or more employees are required to submit the Workforce Monthly Report to the Administrative Division of the Ministry of Work and Social Relations. The report contains workforce information for a given establishment of a company, including

the total number of employees and details of employees who have joined or left the establishment during the month.

Workgroup

A user-defined group of employees who share identical compensation rules. A workgroup may be equivalent to all the employees in a business enterprise, all employees in a Paygroup, all employees belonging to the same Union or Union Local, or all employees who work at a specific work location.

Worklist

The automated "to do" list that Workflow creates. From the Worklist you can directly access the panels you need to perform the next action, and then return to the Worklist for another item.

Worklist

The automated “to do” list which Workflow creates. From the worklist you can directly access the pages you need to perform the next action, and then return to the worklist for another item.

Works Councils (Betriebsrat)

In Germany, the works councils for your company are internal committees elected by the employees that represent the interests of salaried and hourly paid employees, other than management. Every work location in your company has it's works council (this would be the local works council) and the company as a whole has a central works council.

Work-Study Program

Government or non-government programs supervised work experience related to a student's course of study and are a part of, or a supplement to, education. Federal student-trainee programs are examples of such programs.

X

Y

Yearly Maximum Pensionable Earnings (YMPE)

Amount set by the government upon which Canadian Pension Plan (CPP) contributions are made.

Z

Zero-Based Budgeting

A budgeting option that builds a budget from the ground up starting with zero values. This is in contrast to an incremental budget that is based upon using prior year actual or budget values as starting point. PeopleSoft Budgeting-specific.

Zero-Rated VAT

A VAT transaction with a VAT code that has a tax percent of zero. Used to track taxable VAT activity where no actual VAT amount is charged.

Zip Code

The term for postal codes in the United States.

Index

%

%MaxMessageSize( ) 6-19
%TransformData 10-11

A

addConnectorField 13-10
application engine, tracing 9-9
application server
  logging 9-8
  tracing 9-8
APPMSGARCH 8-38
archiving, system messages 8-38
asynchronous messages
  monitoring 8-2, 8-20, 8-28
  understanding 8-2
  viewing errors 8-32
  viewing in XML 8-32
  viewing nonrepudiation information for 8-33
  viewing properties of 8-29
asynchronous publication
  message flow 8-5
  message status 8-5
asynchronous subscription
  message flow 8-8
  message status 8-8
asynchronous transactions 7-17
audit action codes 6-7
authentication 3-5, 7-10

B

brokers 8-2

C

certificate alias 12-46
certificate authentication 7-10
codeset groups 7-12
configuration overview 2-21
  defining local gateways 2-22
  defining remote gateways 2-22
  setting Integration Gateway properties 2-22
  setting up local nodes 2-23
  setting up remote nodes 2-23
connector class development (C/C++ environment) 13-26
  building dlls for native libraries 13-28
  creating Java target connectors 13-27
  creating JNI headers 13-28
  development process 13-26
  implementing JNI header functions 13-28
  registering dlls 13-28
  step-by-step instructions 13-28
connector class development (general)
  development considerations 13-7
  input and output formats 13-7
  installing connector classes 13-18
  local and external system interaction 13-8
  registering connectors 13-18
  reusing connector code 13-31
connector class development (Java XML DOM wrapper) 13-23
  classes 13-24
  code example 13-24
connector class development (listening connectors)
  building error handing into 13-15
  building error logging into 13-15
  controlling message routing 13-13
  development considerations 13-13
  invoking 13-13
  servlet versus non-servlet based 13-13
  template (non-servlet) 13-22
  template (servlet) 13-20
connector class development (target connectors)
  building error handling into 13-11
  building error logging into 13-11
  development considerations 13-8
  infrastructure (introspection) 13-6
  infrastructure (message send) 13-4
  infrastructure (ping) 13-5
  introspection 13-9
  target connector interface 13-8
  template 13-19
connector classes, installing 13-18
connector development 13-3
Connector SDK 13-1
  API documentation 13-3
  contents of 13-2
  location of 13-2
  understanding 13-1
ConnectorDataCollection 13-9
connectors
  editing properties 7-6, 7-16
  in gateway definition 7-5, 7-6
  in node definition 7-14
  JMS 7-4

overview 1-3
registering 7-5
required properties 7-7
contractors 8-2
cookie support, HTTP Listening Connector 12-41
cookies 3-5, 6-14

D

data translation 3-6, 10-20, 11-4
codeset 10-21
codeset group 7-12, 10-20
codeset values 10-21
sample scenario 10-20
XSLT example 10-29
date format 10-3
debugging
quick reference 9-9
subscription PeopleCode 9-9
dispatchers 8-4
changing status of 8-26
status of 8-25
documentation (API) for connector development 13-3

E

EMAIL_ATT_WRK 12-30
EMAIL_REQ_WRK 12-29
EMAIL_RES_WRK 12-29
Encrypting certificate alias password 12-46
error handling
Integration Gateway 9-1
listening connectors 9-2
target connectors 9-1
error notification
messaging system batch processes for 8-36
errors
asynchronous messages 8-32
synchronous messages 8-35
exceptions
Integration Gateway 9-2
Java exceptions 9-4
standard exceptions 9-2
explicit routing 7-10, 7-22
Extensible Stylesheet Language Transformation *See* XSLT

F

FillPubConByChannel() 8-39
FillPubConByMsg() 8-39
FillSubConByChannel() 8-39
FillSubConByMsg() 8-39
filtering 3-6, 10-14, 11-4
Integration Broker Monitor data 8-17
PeopleCode example 10-15
FTP Target Connector
node-level connector properties 12-16
requried JAR files 12-16
using 12-16

G

gateway certificates
encrypting certificate alias password 12-46
setting up for the Integration Gateway 12-45, 12-46
understanding 12-45
gateway definition
JMS connector properties 7-4
local 7-2
refreshing properties 7-4
URL 7-4
Gateway Manager 1-4
gateway services
Connector Management service 1-5
error logging 1-5
message logging 1-5
message validation 1-5
overview 1-4
XML parsing 1-5
gateways
specifying for a node 7-14
generic routing
defined 2-8
hub configuration 2-8

H

handlers 8-4
HTTP Listening Connector
cookie support for 12-41
passing properties to 12-38
properties 12-37
SOAP message support 12-41
using 12-37
HTTP Target Connector
gateway-level connector properties 12-14
node-level connector properties 12-13
using 12-13
hub configurations 2-7
generic-routing 2-8
sender-specific routing 2-10
hub nodes 3-6, 7-12, 11-4, 11-16
retaining messages at 11-16
transaction modifier example 11-16
hub routing types 2-8

I

IBRequest 1-5
IBResponse 1-5
implicit routing 7-10, 7-23
inbound errors 6-24, 6-26, 6-27
inbound messaging 3-3
inbound transactions 7-18
integration
  administrative elements 3-2
  development elements 3-2
Integration Broker Connector SDK *See* connector SDK
Integration Broker Monitor 8-1
  Archive Messages component 8-38
  archiving messages (batch process) 8-38
  error notification (batch processes) 8-36
  Error Notification component 8-36
  extracting information with component interfaces 8-39
  filtering messaging information 8-17
  Message Details component 8-28
  message status 8-13
  Monitor Messages component 8-15
  monitoring asynchronous message details 8-28
  monitoring asynchronous message instances 8-20
  monitoring channel status 8-21
  monitoring messaging system information 8-15
  monitoring publication contracts 8-20
  monitoring subscription contracts 8-21
  monitoring synchronous message details 8-34
  monitoring synchronous message instances 8-21
  monitoring system queue information 8-19
  opening 8-12
  purging messaging tables 8-39
  saving filtering selections 8-18
  setting permissions 8-13
  setting security 8-13
  Synchronous Details component 8-33
  viewing asynchronous message properties 8-29
  viewing asynchronous messages in XML 8-32
  viewing nonrepudiation information for asynchronous messages 8-33
  viewing output 8-19
  viewing synchronous message errors 8-35
  working with node status 8-22
  working with pub/sub server domains 8-24
Integration Engine
  overview 1-2
Integration Engine architecture
  overview 1-6
Integration Gateway
  about setting properties 2-22
  error handling 9-1
  error logging 9-7
  exceptions 9-2
  message and error logging 9-4
  message logging 9-5
  overview 1-1
  posting third-party XML message to 13-56
  proxy authentication 12-48
  running behind a proxy web server 12-48
  setting properties 12-1
  setting up certificates 12-45
  understanding certificates 12-45
  using 12-1
Integration Gateway architecture
  connectors 1-3
  Gateway Manager 1-4
  gateway services 1-4
  overview 1-2
Integration Gateway properties 12-1
  class installation properties 12-2
  connector configuration properties 12-2
  delivered connector configuration properties 12-2
  error logging properties 12-6
  JMS configuration properties 12-8
  Jolt connect strings 12-3
  message logging properties 12-6
  proxy web server properties 12-7
  target connector property (default) 12-3
  version properties 12-2
IntegrationGateway.properties file 12-1
integrations
  understanding 2-1
  with PeopleSoft 8.1 systems 2-20
  with PeopleSoft 8.4 systems 2-2
  with PeopleSoft 8.4 systems using hubs 2-7
  with PeopleSoft 8.4 systems using remote gateways 2-4
  with third-party systems 2-13
  with third-party systems using remote gateways 2-15
IntrospectConnector method 13-9
introspection
  building into target connectors 13-9

J

Java XML DOM wrapper 13-23
  classes 13-24
  code example 13-24
JMS connector properties 12-8
JMS Listening Connector
  gateway-level connector properties 12-45
  message header properties 12-44
  required JAR files 12-45
  shutting down 12-44
  starting 12-44

using 12-43
JMS queue listener 12-43
JMS subscriber 12-43
JMS Target Connector
additional set up steps 12-22
asynchronous and synchronous communication with 12-18
errors and exceptions 12-22
gateway-level connector properties 12-20
message header properties 12-21
node-level connector properties 12-18
point-to-point messaging 12-17
publish and subscribe messaging 12-17
required JAR files 12-21
using 12-17
Jolt connect strings
setting 12-3

K

keystore 12-46

L

language codes 6-7
listening connectors
delivered with Integration Broker 12-36
developing *See* connector class development (listening connectors)
development considerations 13-13
error handling 9-2
HTTP Listening Connector 12-37
JMS Listening Connector 12-43
message flow through on the Integration Gateway 12-36
message logging in 9-6
overview 1-4
PeopleSoft 8.1 Listening Connector 12-42
PeopleSoft Listening Connector 12-42
understanding 12-36
local gateways, about defining 2-22
local nodes
about setting up 2-23
adding pause times for 8-22
deleting pause times for 8-23
scheduling system pause times for 8-22
testing 8-23
logging
application server 9-8
error logging 9-4
error logging methods and parameters 9-8
gateway error logging 9-7
gateway message logging 9-5
message logging 9-4
message logging methods and parameters 9-6
logging synchronous transactions 7-25
logMessage 9-6

M

master nodes 7-12
message attributes *See* PSCAMA
message channel
activating 4-4
blocked 8-14
defining 4-1
monitoring 8-21
stalled 8-15
message compression and encoding 12-10
message definition
accessing PeopleCode 5-11
field alias 5-4
format types 5-1
list view 5-3
modifying underlying records 5-7
OnRequest event 5-15
opening from channel definition 4-3
reorganizing 5-8
structure view 5-3
subscriptions 5-12
views 5-3
message errors page in message monitor 8-32
message format
example 10-5
FieldTypes section 10-3
MsgData section 10-4
non-rowset XML 3-5
non-XML 3-5, 6-10, 10-13
rowset based 6-1, 10-3
SOAP compliant 3-5, 6-3
types 5-1
XML DOM compliant 3-5, 6-2
message monitor
errors page 8-32
XML message viewer 8-32
message nodes
monitoring 8-22
message processing
on remote gateways 2-17
testing with Send Master 13-33
message server domains
inactivating 8-26
message server processes 8-3
PSBRKDSP 8-4
PSBRKHND 8-4
PSPUBDSP 8-4
PSPUBHND 8-4
PSSUBDSO 8-4
PSSUBHND 8-4
message status 8-13
message system reports, generating with queries 8-27
message tables

purging 8-39
message validation 1-5
message versions 3-4
creating 5-5
default structure 5-6
inserting records 5-6
setting a default 5-10
messages
activating 5-9
archiving 4-4, 8-38
assigning to message channel 4-3, 5-9
cookies in 6-14
date format 10-3
external name 7-25
processing order 4-5, 6-9
reply 6-5
request 6-4, 6-11, 7-20
response 6-4, 6-11, 7-25, 11-14
undefined 4-6
viewing and correcting 5-10, 6-28
monitoring
asynchronous message instances 8-20
channels 8-21
message nodes 8-22
messaging system information 8-15
publication contracts 8-20
subscription contracts 8-21
synchronous messages 8-21
MSGSTATUSSUMMARY 8-39

N

node definition 7-9, 7-12
authentication 7-10
editing connector properties 7-16
hub 7-12
master 7-12
optional properties 7-13
routing type 7-10
specifying a connector for 7-14
specifying a gateway for 7-14
nodes
activating 7-11
default local 7-11
local 7-8, 7-11
remote 7-8
type 7-9
using as hubs 3-6, 11-4
nonrepudiation 3-5, 5-9, 5-10, 7-11, 10-7

O

outbound messaging 3-3
outbound transactions 7-17

P

password authentication 7-11
pause times 8-22
adding for local nodes 8-22
deleting for local nodes 8-23
PeopleBooks
printed, ordering ii
PeopleCode 6-9, 10-10
accessing 5-11
debugging subscription PeopleCode 9-9
EditError property 6-25
ExecuteEdits method 6-24
Exit function 6-25
inbound asynchronous 6-15
inbound synchronous 6-22
Message class example 6-10, 6-12, 6-15, 6-16, 6-18, 6-19, 6-22
outbound asynchronous 6-10
outbound synchronous 6-11
SoapDoc class example 6-13, 6-23
TransformData class 10-12
XmlDoc class example 6-11, 6-12, 6-20, 6-22
PeopleSoft 8.1 Listening Connector
using 12-42
PeopleSoft 8.1 systems
integrations with 2-20
PeopleSoft 8.1 Target Connector
gateway-level connector properties 12-15
node-level connector properties 12-15
using 12-15
PeopleSoft Common Application Message Attributes *See* PSCAMA
Peoplesoft Listening Connector
using 12-42
Peoplesoft Target Connector
using 12-15
PeopleSoft Target Connector
gateway-level connector properties 12-15
node-level connector properties 12-15
Ping method 13-9
pinging
remote nodes 8-24
POP3 services 12-23
POP3 Target Connector
default node, setting up 12-27
default node, using 12-28
EMAIL_ATT_WRK record 12-30
EMAIL_REQ_WRK record 12-29
EMAIL_RES_WRK record 12-29
gateway-level connector properties 12-27
node-level connector properties 12-25
passing node-level connector properties 12-26
POP3 services 12-23
PT_EMAIL_POP3 12-27
retrieving specific messages from the POP3 server 12-30

using 12-23
posting third-party XML messages
to the Integration Gateway 13-56
private key 12-46
properties, Integration Gateway 12-1
proxy authentication 12-48
proxy web server
running the Integration Gateway behind 12-48
proxy web server properties 12-7
PSBRKDSP 8-4
PSBRKHND 8-4
PSCAMA 6-5
audit action codes 6-6, 6-7
inclusion in a message 6-8
language codes 6-6, 6-7
message sequence 6-6
psft_function node *See* data translation. *See* XSLT, psft_function node
PSPUBDSP 8-4
PSPUBHND 8-4
PSSUBDSP 8-4
PSSUBHND 8-4
PT_AMM_WF 8-36
PT_EMAIL_POP3 12-27
using 12-28
pub/sub server domains
working with 8-24
public key 12-46
publication
synchronous message flow and message status 8-10
Publication Broker 8-3
Publication Contractor 8-3
publication contracts
monitoring 8-20

Q

queries
running to generate message system reports 8-27
queues 8-2

R

relationship definition
overriding the contact information 11-8
overriding the node properties 11-8
specifying node pairs 11-6, 11-7
relationships 3-5, 11-1
activating 11-6, 11-7
transaction modifiers in *See* transaction modifiers
when to use 11-1
where to define 11-4, 11-15
remote gateways
about defining 2-22
flow of messages through 2-17
message processing on 2-17
sending messages to third-party systems 2-15
remote nodes
about about setting up 2-23
remote nodes, pinging 8-24
routing type 7-10, 7-22

S

Send Master 13-33
Connector Introspection window 13-46
groups of projects, creating 13-53
groups of projects, managing 13-53
groups of projects, testing 13-54
headers used in 13-41
Input File projects 13-42
Integration Broker projects 13-44, 13-49
opening 13-34
project types 13-40
sharing projects and groups 13-55
understanding 13-33
viewing output 13-54
workspaces 13-35
Send method 13-9
sender-specific routing
defined 2-8
hub configuration 2-10
Simple File Target Connector
node-level connector properties 12-35
using 12-35
Simple Post Tool 13-56
parameters 13-58
setting environment variables 13-57
software requirements 13-57
syntax 13-57
usage 13-57
using 13-58
SMTP Target Connector
node-level connector properties 12-23
using 12-22
SOAP message support
HTTP Listening Connector 12-41
subscription
synchronous message flow and message status 8-11
subscription contract status 6-27
Subscription Contractor 8-3
subscription contracts
monitoring 8-21
synchronous messages
monitoring 8-21
monitoring details 8-34
understanding 8-10
viewing errors 8-35
synchronous publication

message flow and message status 8-10
synchronous subscription
message flow and message status 8-11
synchronous transactions 7-17
logging 7-25
multiple target nodes 7-21
system queue information
monitoring 8-19
system variables
%TransformData 10-11

T

target connectors
delivered with Integration Broker 12-12
developing *See* connector class development (target connectors)
development considerations 13-8
error handling 9-1
FTP Target Connector 12-16
gateway-level connector properties 12-13
HTTP Target Connector 12-13
JMS Target Connector 12-17
message flow through on the Integration Gateway 12-11
message logging in 9-5
node-level connector properties 12-12
overview 1-3
PeopleSoft 8.1 Target Connector 12-15
PeopleSoft Target Connector 12-15
POP3 Target Connector 12-23
Simple File Target Connector 12-35
SMTP Target Connector 12-22
understanding 12-11
understanding properties of 12-12
templates for connector development 13-18
testing
local nodes 8-23
message processing (Send Master) 13-33
testing a message 5-16, 6-9
third-party systems
integrations with 2-13
integrations with using remote gateways 2-15
sending messages to PeopleSoft 8.4 systems from 2-17
sending messages to using remote gateways 2-15
tracing
application engine 9-9
application server 9-8
transaction definition 7-19
external message name 7-25
message detail 7-24
overriding the connector 7-23, 7-25
overriding the gateway 7-23
overriding the routing type 7-22
request message 7-20
response message 7-25
transaction types 7-20
transaction modifiers 3-5, 11-2
activating 11-14
applying transform programs with 11-4, 11-14
asynchronous to synchronous 11-3, 11-14, 11-15
defining 11-10
inbound to outbound 11-3
narrowing parameter choices for 11-12
outbound to inbound 11-3
selecting transaction types for 11-11, 11-13
transaction type combinations 11-2
using for hub routing 11-3, 11-16
transactions
activating 7-22, 7-23
for cross-node messaging 7-18
for local messaging 7-18
inbound asynchronous 7-20
inbound synchronous 7-20
modifying 3-5
outbound asynchronous 7-20
outbound synchronous 7-20
transmission types 3-6
types 6-4, 7-17
viewing undelivered node transactions 8-24
transform programs 3-6, 10-8, 10-10
filtering messages with 10-14
handling non-XML data 10-13
invoking 10-11, 11-14
limitations 10-2, 10-8
reporting errors 10-17
supported languages 10-10
tracing 10-11
transforming messages with 10-17
translating data with 10-20
working data 10-13
transformation 3-6, 10-17, 11-4
XSLT example 10-18
transmission types 11-3

U

UIDL 12-23
upgrade issues 4-5, 5-3
uppercasing inbound data 5-7

V

validation
rowset data 6-24
SOAP compliance 6-24
XML DOM compliance 6-24

X

XML message viewer 8-32
XML parsing 1-5
XSLT 10-10
applying transformation with 10-18
performing data translation with 10-27
psft_function node 10-27
psft_function parm tag 10-28
psft_function value tag 10-28